UNIVERSITE SIDI MOHAMED BEN ABDELLAH
FACULTE DES SCIENCES ET TECHNIQUES FES
DEPARTEMENT D'INFORMATIQUE

Projet de Fin d'Etudes

Licence Sciences et Techniques Génie Informatique

Développement d'une application WEB pour la gestion du stock dans un établissement universitaire

---

Lieu de stage : La société **T-CREATIVE**

Réalisé par :

- ASBAYO Hamza
- SALIM Amine

Encadré par :

- Pr. ZAHI Azeddine

Soutenu le 12/06/2019 devant le jury composé de :

Pr. MAJDA Aicha

Pr. ZAHI Azeddine

Pr. BEN ABBOU Rachid

Année Universitaire 2018-2019

Développement d'une application WEB pour la gestion du stock dans un établissement universitaire

Remerciements

Avant tout développement sur cette expérience professionnelle, il apparait opportun de commencer ce rapport de stage par des remerciements.

Nous tenons à remercier dans un premier temps, toute l'équipe pédagogique de la faculté des sciences et techniques et les intervenants professionnels responsables de la LST génie informatique qui ont contribué à nous donner une formation solide tout au long de nos années d'études.

Nous remercions tout particulièrement notre encadrant académique Mr. ZAHI Azeddine, pour son encadrement, tous ses conseils, le soutien dont il nous a fait bénéficier, de ses encouragements, les aides précieuses qu'il n'a pas cessés de nous apporter tout au long de la période de ce projet et pour ses efforts afin d'assurer le bon déroulement de ce stage.

Nous tenons également à remercier et à témoigner toute notre reconnaissance à Mr. ALAMI Ibrahim, président de l'association, pour son accueil et la confiance qu'il nous a accordée dès notre arrivée dans l'association ; pour le temps qu'il nous a consacré tout au long de cette période, sachant répondre à toutes nos interrogations ; sans oublier sa participation au cheminement de ce rapport.

Nous remercions également Pr. BEN ABBOU Rachid, Pr. ZAHI Azeddine et Pr. MAJDA Aicha, les membres du jury pour nous avoir honoré en acceptant d'évaluer et de juger ce travail.

Enfin nous remercions toutes les personnes qui ont contribuées de près ou de loin à la réalisation de ce travail, ainsi qu'au bon déroulement du stage, et dont les noms ne figurent pas dans ce document.

Résumé

Le projet réalisé dans ce stage vise à développer une application web facilitant la gestion de stock de ressources et des équipements des établissements publics.

Cette application permettra l'automatisation du processus de demande des fournitures, la gestion des profils des utilisateurs ainsi que la gestion des rapports.

En vue d'appliquer les acquis du cours Modélisation en UML, nous avons fait tout d'abord une étude conceptuelle de l'application. Cette dernière nous a facilité la réalisation de l'application en organisant les idées et en structurant l'implémentation suivant des diagrammes. L'application a été implémentée par diverses technologies en se basant sur l'étude conceptuelle et a été implémentée avec Laravel, JQuery et Bootstrap, qui sont des Framework permettant de créer efficacement des sites web complexes et flexibles.

En effet, ce stage est considéré comme une occasion qui nous a permis d'entrer en contact direct avec le monde du travail et nous a premièrement aidé à renforcer la théorie par la pratique, deuxièmement à élargir nos connaissances, et enfin à s'adapter à la vie professionnelle et établir des rapports directs avec les autres, ce qui débouche sur une intégration assez rapide et favorable.

Abstract

The project carried out in this internship aims to develop a web application facilitating the management of stock resources and equipment of public institutions. This application will automate the process of requesting supplies, managing user profiles, as well as reports.

In order to apply the knowledge acquired in the Modeling course in UML, we first made a conceptual study of the application, which facilitated the realization of the application by organizing the ideas and structuring the implementation according to diagrams. The application has been implemented by various technologies based on the concept study and has been implemented with Laravel, JQuery, and Bootstrap, which are frameworks for creating complex and flexible websites.

Indeed, this internship is considered an opportunity that allowed us to get in touch with the world of work and helped us first to strengthen the theory through practice, second to expand our knowledge, and finally to adapt to professional life and establish direct relationships with others, leading to a fairly rapid and favorable integration.

Table des matières

Remerciements 3
Résumé 4
Abstract 4
Table des matières 5
Liste des figures 7
Liste des tableaux 8
Liste des abréviations 9
Introduction 10
Chapitre I : Contexte générale du projet 11
1. Organisme d'accueil : 12
1.1. Présentation et Organigramme 12
1.2. Services et activités de T-CREATIVE 13
2. Présentation du projet 14
2.1. Contexte et Objectifs 14
2.2. Cahier des charges 16
2.3. Conduite du projet 17
2.4. Planning du projet 18
Chapitre II : Analyse et conception 19
1. Analyse des besoins 20
1.1. Les besoins fonctionnels 20
1.2. Les besoins non fonctionnels 21
2. Méthodes d'analyse et conception 22
2.1. Le langage UML 22
2.2. Le model MVC 22
2.2.1. Définition 22
3. Diagrammes de cas d'utilisation 24
3.1. Identification des acteurs et leurs rôles 24
3.2. Les messages émis et reçus 27
3.3. Diagramme de cas d'utilisation relatif au responsable du stock 30
3.4. Diagramme de cas d'utilisation relatif aux enseignants 31
3.5. Diagramme de cas d'utilisation relatif au chef de département 31

4. Diagrammes de séquence 32
5. Diagramme de classes 38
6. Modèle logique de données 40
Chapitre III : Mise en œuvre du projet 41
1. Outils de développement 42
1.1. Technologies Web 42
1.2. Les Framework de développement web 43
1.3. Choix technologiques 44
2. Présentation de l'application 48
2.1. Interfaces du responsable du stock 50
2.2. Interfaces des enseignants 56
2.3. Interfaces du chef de département 58
Conclusion et perspectives 60
Webographie 61

Liste des figures

Figure 1 : Organigramme de la société T-CREATIVE....12
Figure 2 : le cycle de développement en Y....17
Figure 3 : Diagramme de Gantt....18
Figure 4 : L'architecture MVC....23
Figure 5 : Diagramme de cas d'utilisation relatif au responsable du stock....30
Figure 6 : Diagramme de cas d'utilisation relatif à l'enseignant....31
Figure 7 : Diagramme de cas d'utilisation relatif au chef de département....31
Figure 8 : Diagramme de séquence - Authentification -....33
Figure 9 : Diagramme de séquence - Ajouter Article -....34
Figure 10 : Diagramme de séquence - Effectuer demande -....35
Figure 11 : Diagramme de séquence - Gestion demandes -....36
Figure 12 : Diagramme de séquence - Gestion réclamations -....37
Figure 13 : Diagramme de classes....39
Figure 14 : Modèle logique de données....40
Figure 15 : schéma d'exécution d'un script avec le pattern MVC de Laravel....45
Figure 16 : statistiques des 15 Meilleurs Framework PHP gratuits....46
Figure 17 : Fenêtre d'authentification....49
Figure 18 : Message d'erreur - utilisateur introuvable -....49
Figure 19 : Fenêtre d'accueil du responsable de stock....50
Figure 20 : Fenêtre de listes des articles....50
Figure 21 : Formulaire d'ajouter un article....51
Figure 22 : Message de confirmation -Ajouter article –....51
Figure 23 : Les articles en état critique....51
Figure 24 : Imprimer rapport PDF....52
Figure 25 : Fenêtre de gestion des utilisateurs....52
Figure 26 : Ajouter un nouvel utilisateur....53
Figure 27 : Fenêtre de gestion des demandes....53
Figure 28 : Afficher les détails d'une demande....54
Figure 29 : Confirmer la suppression....54
Figure 30 : Fenêtre de gestion des réclamations....55
Figure 31 : Afficher les détails d'une réclamation....55
Figure 32 : Effectuer une demande - interface1 –....56
Figure 33 : Effectuer une demande – interface2 -....56
Figure 34 : Message d'erreur – Quantité indisponible -....57
Figure 35 : Liste des demandes....57
Figure 36 : Notification - Article commandé est disponible -....57
Figure 37 : Fenêtre d'accueil du chef de département....58
Figure 38 : Fenêtre - Sélectionner option -....58
Figure 39 : Liste des demandes du département....59
Figure 40 : Liste des réclamations du département....59

Liste des tableaux

Tableau 1 : Les acteurs et leurs rôles.....26
Tableau 2 : Les messages émis et reçus 29

Liste des abréviations

Abréviation	Désignation
HTML	Hypertext Markup Language
PHP	*Hypertext Preprocessor*
CSS	Cascading Style Sheets
SQL	Structured Query Language
SGBD	Système de Gestion de Base de Données
MVC	Model View Controller
UML	Unified Modeling Language
POO	Programmation Orientée Objet

Introduction

Pour acquérir une bonne et parfaite qualité, la formation théorique seule ne suffit pas, il est donc nécessaire de suivre une démarche réelle permettant de voir comment se déroulent les tâches dans la vie professionnelle.

A cet effet, afin de valider nos études acquises au fil de trois ans au sein de Faculté des Sciences et Techniques de Fès et en vue de l'obtention de la Licence spécialisée en informatique, nous sommes amenés à effectuer un stage de fin d'études de huit semaines, que nous avons effectué dans la société T-CREATIVE, spécialisée dans la création des applications et sites web ; Notre mission dans ce stage était de mettre en place une solution informatisée sous forme d'une application web dédié à la gestion du stock des établissements publiques (universités, écoles…).

Dans le cadre de ce projet de fin d'études, on va essayer de répondre à ces objectifs tout en appliquant les notions théoriques qu'on a vues au cours de notre formation à la FSTF. On va suivre une approche orienté objet de conception suivant la modélisation UML, et utiliser les différentes techniques de web adéquates pour la réalisation de ce projet.

Dans ce rapport que vous feuilletez, nous présenterons les étapes que nous avons suivis pour réaliser cette application. Ce rapport comporte trois parties :

Dans la première partie nous présenterons, en premier lieu l'organisme d'accueil pendant la durée de stage. En deuxième lieu, nous déterminons les différents objectifs de l'application sous forme d'un cahier de charge bien structuré, ainsi que le plan suivi.

La deuxième partie aborde la phase de conception de données. Nous spécifierons la méthodologie adaptée pour la conception, après, nous présenterons les acteurs et les différents diagrammes et nous détaillerons les différents cas d'utilisation de l'application.

Finalement, au niveau du dernier chapitre intitulé réalisation, nous présenterons les différents outils de développement, ainsi que les principales interfaces graphiques réalisées et quelques scénarios applicatifs.

Chapitre I

Contexte générale du projet

1. Organisme d'accueil :

1.1. Présentation et Organigramme

T-CREATIVE est une société informatique située sur la ville de Fès. Fondée en 2010, elle s'intéresse aux métiers de communication, au développement d'applications de gestion et de sites web tels que la création *graphique*, sites *e-commerce*.

Issue d'une passion, T-CREATIVE offre aussi à ses clients une multitude de services. En effet, l'équipe T-CREATIVE conseille et concrétise les idées de ses clients en leurs offrant des solutions sur mesure : des créations graphiques uniques et personnalisées pour l'identité visuelle (logo), l'affichage urbain, brochures, dépliants et catalogues produits, ainsi que la conception ou la refonte des sites web de tous types : institutionnel, catalogues produits avec CMS et sites marchands, des newsletters et des campagnes d'e-mailing et la publicité sur Facebook, etc. En plus, de ces service, T-CREATIVE met à la disposition des clients un service d'achat de nom de domaine et l'hébergement sur des serveurs avec la configuration des comptes messageries professionnels.

L'organigramme de l'entreprise repose sur les différentes fonctions exercées au sein de l'organisation (Développement, commerce, réseau et maintenance, multimédia…).

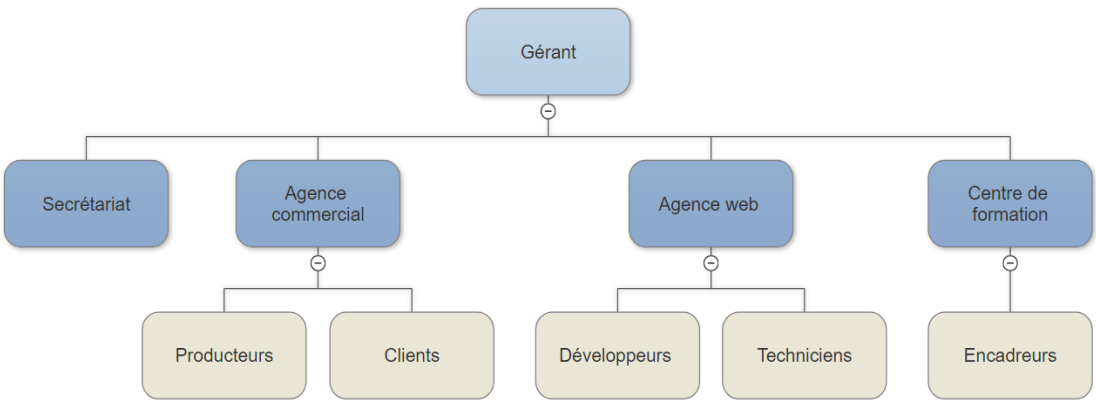

Figure 1 : Organigramme de la société T-CREATIVE

1.2. Services et activités de T-CREATIVE

L'agence T-CREATIVE propose une large gamme de services tel que :

- ***Développement Web* :** De nos jours, les sites Internet jouent un rôle primordial dans les activités de vente et de présentation des produits au sein des entreprises. En effet, un site Internet offre des outils efficaces pour maintenir une relation permanente avec ses clients. C'est pour cela que l'agence web porte ses solutions web pour créer tout type de sites : site vitrine, site e-commerce, site dynamique, site statique, site flash, site administrable, ou site de formation.

- ***Hébergement* :** T-CREATIVE héberge également les sites web. Les packs d'hébergements contiennent les caractéristiques suivantes : (les caractéristiques dépendent des packs choisis)

  - ✓ Nom de domaine
  - ✓ Espace disque allant jusqu'à 3000MB
  - ✓ Transfert par mois allant jusqu'à 500Go
  - ✓ Comptes Email
  - ✓ Réponse automatique
  - ✓ Protection anti-virus
  - ✓ Base de données MySQL

- ***Design et création graphique :*** Les infographies, sont les premiers éléments sur lesquels un lecteur pose son regard. T-CREATIVE assure la réalisation d'illustrations très diverses (plaquettes, flyers, cartes de visites, création de logos, journaux d'entreprise, affiches…) et cela comporte :
  - ***La création de logo et d'identité visuelle* :** Le logo est l'élément fondateur de l'identité visuelle, facteur essentiel de la réussite de la réussite d'une entreprise ou d'une marque. Le studio de T-CREATIVE élabore avec le client les principales idées que doit véhiculer l'image de sa société, ses valeurs, son secteur et ses ambitions. Une fois crée, le logo peut être décliné sur tous les supports de communication du client (papier à en-tête, cartes de visite, enveloppes, site internet, signalétique, véhicules…).

- ***La création de charte graphique :*** En complément du logo, T-CREATIVE réalise une charte graphique cohérente et flexible. Elle déterminera le style graphique de chacun des supports de communication et garantira l'homogénéité de l'identité visuelle du client.

- ***La création de packaging*** : L'emballage est l'un des principaux arguments de séduction d'un produit. Le studio de création de T-CREATIVE conçoit pour le packaging, une identité graphique originale, forte et séduisante dans le respect des standards du produit et secteur du client.

- ***Photographie et vidéo*** **:** Professionnalisme et expérience pour illustrer tous les supports, plaquettes publicitaires, catalogues papier et catalogues électronique, vidéos d'entreprise ou encore sites internet. T-CREATIVE s'est dotée des équipements photographiques les plus modernes servis par de vrais professionnels. Le studio photo de T-CREATIVE s'est adapté aux évolutions technologiques. Elle propose également un outil de dernière génération pour les packshots destinés à illustrer les produits et réalise des prises de vues à 360°.

- ***Centre de formation :*** Un centre de formation informatique qui propose des formations divisées en 2 parties :
  - *La formation logicielle* : bureautique, développement web, Référencement SEO, MultiMedia…
  - *La formation générale* : comptabilité, langues étrangères, management, commercialisation…

2. Présentation du projet

2.1. Contexte et Objectifs

Dans le cadre de notre stage au sein de la société T-CREATIVE, l'équipe nous a confié un projet de développement web. Il s'agit de concevoir et réaliser une application web générique de gestion de stock dédiée aux établissements universitaires privés et publics. Ce premier projet, lancé par la société d'accueil, a pour objectif d'offrir une solution informatique de gestion de stock adaptée aux établissements universitaires. Les motivations principales qui ont conduit à l'initiation de ce projet sont :

- L'inadéquation des outils disponibles sur le marché aux établissements universitaires. En effet, les outils disponibles sont destinés aux grandes entreprises qui ont un processus de gestion différent.

- L'absence d'outils informatique efficace pour la gestion de stock pour les établissements universitaires. En effet, la plupart des établissements publics, notamment privés, gèrent leurs stocks d'une manière artisanal à l'aide de support papiers et de fichiers Excel. Cela conduit à une inefficacité qui peut impacter le déroulement des autres activités au sein de l'établissement.

Ainsi notre mission dans ce projet est de fournir une application web qui implémente le processus de gestion de stock dans un établissement universitaire. L'application doit offrir aux personnels de l'établissement une solution automatique et facile pour réaliser les activités suivantes :

- Gestion des demandes faites par le personnel concernant les ressources nécessaires.
- Gestion des réclamations sur les ressources acquises et les demandes effectuées.
- Suivi des demandes des ressources et des réclamations.
- Gestion des entrées et sorties du stock.

2.2. Cahier des charges

Avant de commencer la réalisation de notre projet, nous avons procédé à l'élaboration du cahier de charges de notre application. Pour ce faire, nous avons effectué plusieurs réunions avec l'équipe de développement de la société d'accueil. Nous avons ainsi identifié les acteurs concernés et les fonctionnalités que le système doit leurs offrir.

Les acteurs :

- Le responsable du stock (magasinier),
- Le chef de département ou de service,
- Les enseignants et les administrateurs.

Les fonctionnalités regroupées par acteurs :

- *Responsable du stock (magasinier).* L'application doit offrir au magasinier de gérer :
    - Les ressources.
    - Les catégories.
    - Les demandes.
    - Les réclamations.
    - Les utilisateurs.

- *Enseignants ou administratifs.* Le système doit permettre aux enseignants de :
    - Effectuer une demande.
    - Faire une réclamation.

- *Chef de département.* Le système doit permettre au chef de département la possibilité de :
    - Visualiser les demandes et les réclamations des enseignants appartenant à son département.
    - Listage des ressources.
    - Effectuer une demande.
    - Faire une réclamation.

2.3. Conduite du projet

Pour réaliser notre projet, nous avons choisi de travailler avec le processus de développement en Y qui est basé sur deux aspects : fonctionnel et technique (figure 2).

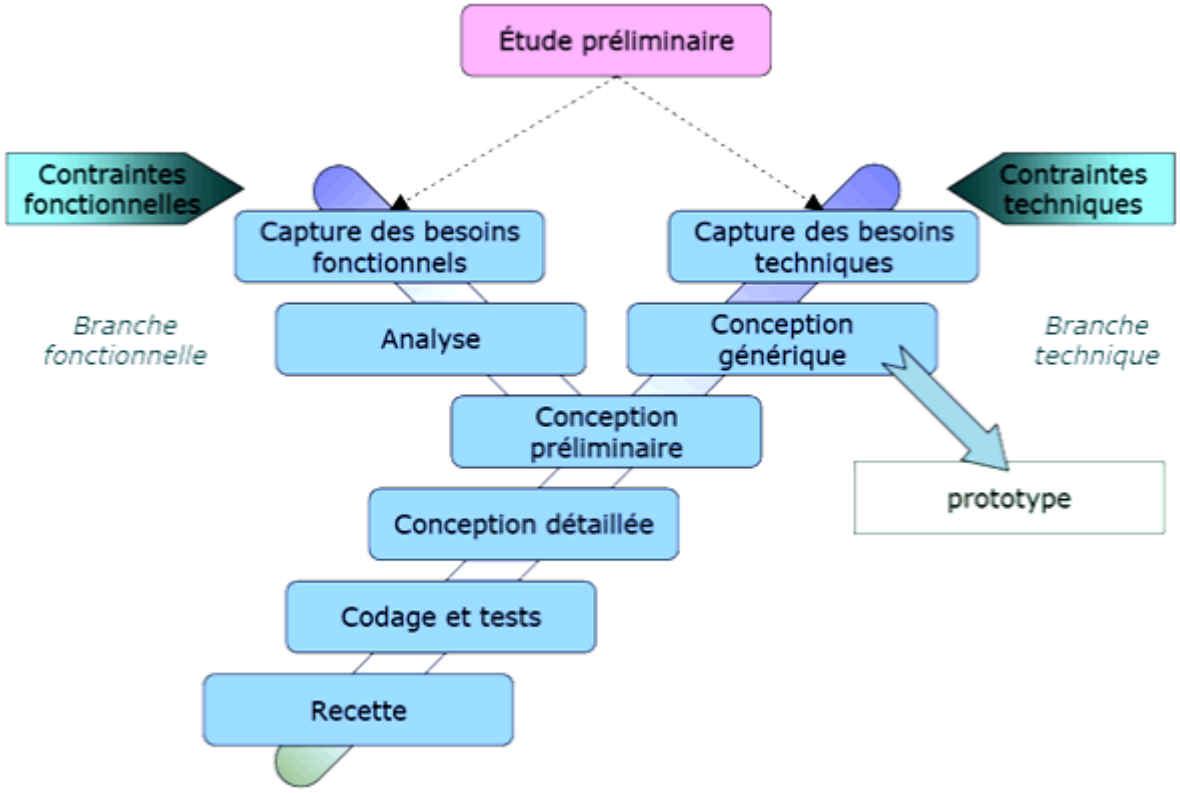

Figure 2 : le cycle de développement en Y

La branche gauche (fonctionnelle) comporte

- La capture des besoins fonctionnels, qui produit un modèle des besoins focalisé sur le métier des utilisateurs. Elle qualifie au plus tôt le risque de produire un système inadapté aux utilisateurs. De son côté, la maîtrise d'œuvre consolide les spécifications et en vérifie la cohérence et l'exhaustivité.
- L'analyse, qui consiste à étudier précisément la spécification fonctionnelle de manière à obtenir une idée de ce que va réaliser le système en termes de métier. Les résultats de l'analyse ne dépendent d'aucune technologie particulière.

La branche droite (architecture technique) comporte

- La capture des besoins techniques, qui recense toutes les contraintes et les choix dimensionnant la conception du système. Les outils et les matériels sélectionnés ainsi que la

prise en compte de contraintes d'intégration avec l'existant Conditionnent généralement des prérequis d'architecture technique.

- La conception générique, qui définit ensuite les composants nécessaires à la construction de l'architecture technique. Cette conception est complètement indépendante des aspects fonctionnels. Elle a pour objectif d'uniformiser et de réutiliser les mêmes mécanismes pour tout un système. L'architecture technique construit le squelette du système informatique et écarte la plupart des risques de niveau technique. L'importance de sa réussite est telle qu'il est conseillé de réaliser un prototype pour assurer sa validité.

La branche du milieu comporte

- La conception préliminaire, qui représente une étape délicate, car elle intègre le modèle d'analyse dans l'architecture technique de manière à tracer la cartographie des composants du système à développer.
- La conception détaillée, qui étudie ensuite comment réaliser chaque composant.
- L'étape de codage, qui produit ces composants et teste au fur et à mesure les unités de code réalisées.
- L'étape de recette, qui consiste enfin à valider les fonctions du système développé.

2.4. Planning du projet

La planification du projet consiste à prévoir le déroulement des tâches tout au long des phases constituant le cycle de développement. Pour schématiser le planning du projet, le choix a été porté sur l'outil *GanttProject* pour sa réputation parmi les logiciels libres de gestion de projet. Il permet la planification d'un projet à travers la réalisation d'un diagramme de Gantt. La figure 3 montre le planning de réalisation de notre projet.

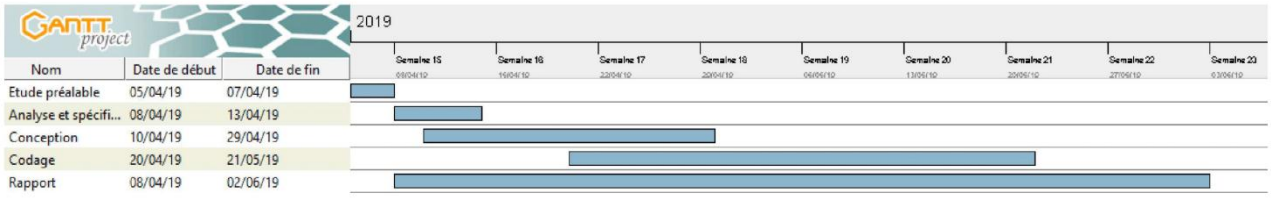

Figure 3 : Diagramme de Gantt

Chapitre II

Analyse et conception

1. Analyse des besoins

1.1. Les besoins fonctionnels

Les besoins fonctionnels sont les besoins spécifiant un comportement d'entrée/sortie du système, l'application doit permettre de :

- Gérer les utilisateurs du système
  - Visualiser les utilisateurs.
  - Chercher à un utilisateur.
  - Ajouter un utilisateur.
  - Modifier un utilisateur.
  - Supprimer un utilisateur.

- Gérer le stock
  - Consulter les articles du stock.
  - Chercher un article.
  - Ajouter un article.
  - Modifier un article.
  - Supprimer un article.

- Gérer les catégories
  - Lister les catégories.
  - Chercher une catégorie.
  - Ajouter une catégorie.
  - Modifier une catégorie.
  - Supprimer une catégorie.

- Gérer les demandes
  - Lister les demandes.
  - Vérifier les demandes (Accepter/Refuser).
  - Imprimer le reçu de décharge.

- Gérer les réclamations
  - Consulter les réclamations.
  - Vérifier les réclamations.

- Effectuer des demandes.
- Faire des réclamations.
- Suivre l'état de ses demandes/réclamations.

1.2. Les besoins non fonctionnels

Pour mettre en place une solution adéquate, on doit prendre en considération les contraintes suivantes :

- **Contrainte sur l'application**

Ces besoins concernent généralement la rapidité et la capacité d'exécution des différentes opérations de l'application. Dans ce sens, elle doit se caractériser par la fiabilité, la rapidité d'ajout, modification et de suppression des données, la rapidité lors de l'exécution des opérations (authentification, ajout, suppression, recherche...).

- **Contrainte ergonomique**

En réalisant notre application, le point le plus important est la satisfaction de l'utilisateur, on vise souvent à faciliter la réalisation de certaines tâches qui sont très importantes pour l'organisation du travail. En effet, notre application doit inclure des interfaces graphiques claires et communes ce qui facilitera par la suite la réalisation de ces tâches. Dans ce sens, ces fonctionnalités doivent être hiérarchisées selon des objectifs principaux à menus à partir desquels on peut accéder à ces interfaces afin d'effectuer des opérations.

- **Critères d'évaluation**

  - *Charge de travail :* Plus les éléments présentés à l'écran sont compréhensibles, plus l'attention nécessaire pour comprendre et maîtriser la navigation est réduite, plus l'interaction avec l'application est rapide et efficace, et moins il y a de risques d'erreur ou d'égarement.
  - *Gestion des erreurs :* rédiger avec soin les messages d'erreur qui doivent être clairs, indiquant une solution possible.

2. Méthodes d'analyse et conception

2.1. Le langage UML

Dans le cadre de notre projet on a choisi UML pour une modélisation objet et qui est l'un des caractéristiques du processus Y

- **Justification du choix d'UML**

  - UML est avant tout un support de communication performant, qui facilite la représentation et la compréhension de solutions objet.
  - L'aspect formel de sa notation, limite les ambiguïtés et les incompréhensions.
  - Son indépendance par rapport aux langages de programmation, aux domaines d'application et aux processus, en fait un langage universel.
  - UML contrairement à son prédécesseur MERISE qui est une méthode systémique (Orienté Donnée), donne un sens intéressant à l'approche objet et couvre de plus tout le cycle de réalisation du logiciel.
  - Il cadre l'analyse.
  - Il permet également de générer automatiquement une partie de code, par exemple en langage Java, grâce aux outils de modélisation UML.

2.2. Le model MVC

2.2.1. Définition

L'architecture MVC (modèle, vue et contrôleur) est un concept très puissant qui intervient dans la réalisation d'une application. Son principal intérêt est la séparation des données (modèle), de l'affichage (vue) et des actions (contrôleur), ce qui assure la clarté de l'architecture et simplifie la tâche du développeur responsable de la maintenance et de l'amélioration du projet.

Les différentes interactions entre le modèle, la vue et le contrôleur sont résumées par le schéma de la figure :

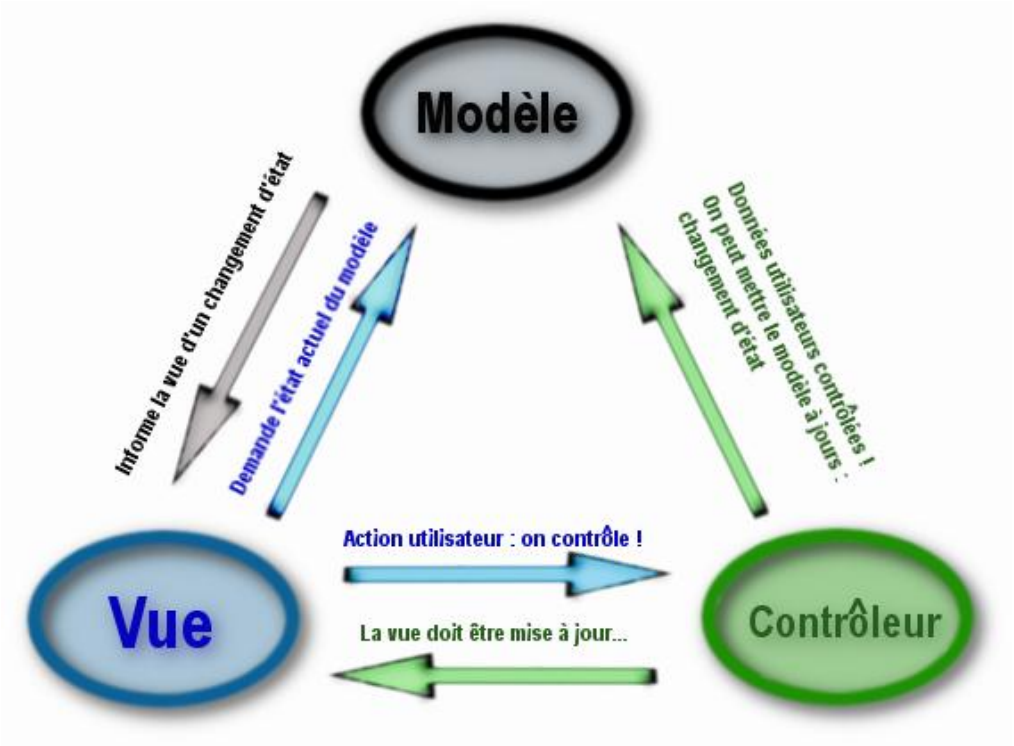

Figure 4 : L'architecture MVC

- **Le Modèle :** Le modèle représente le cœur de l'application : traitements des données, interactions avec la base de données. Il décrit les données manipulées par l'application. Il regroupe la gestion de ces données et, il est responsable de leur intégrité. La base de données sera l'un de ses composants. Le modèle comporte des méthodes standards pour mettre à jour ces données (insertion, suppression, changement de valeur). Il offre aussi des méthodes pour récupérer ces données. Les résultats renvoyés par le modèle ne s'occupent pas de la présentation, Le modèle ne contient aucun lien direct vers la vue.
- **La Vue :** C'est avec quoi l'utilisateur interagit se nomme précisément la vue. Sa première tâche est de présenter les résultats renvoyés par le modèle, sa seconde tâche est de recevoir toute action de l'utilisateur (clic de souris, sélection d'un bouton radio, coche d'une case, entrée de texte, de mouvements, de voix, etc..). Ces différents événements sont envoyés au contrôleur. La vue n'effectue pas de traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur.
- **Le Contrôleur :** Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et déclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle et ce dernier notifie la vue que les données ont changée pour qu'elle se mette à jour. D'après le

patron de conception observateur/observable, la vue est un « observateur » du modèle qui est « observable ». Certains événements de l'utilisateur ne concernent pas les données mais la vue. Dans ce cas, le contrôleur demande à la vue de se modifier. Le contrôleur n'effectue aucun traitement, ne modifie aucune donnée, il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondant à la demande.

- **Justification de choix du modèle MVC**

On a choisi le MVC comme méthodologie d'analyse pour les raisons suivantes :

- Une conception claire et efficace grâce à la séparation des données de la vue et du contrôleur.
- Un gain de temps de maintenance et d'évolution du site.
- Une plus grande souplesse pour organiser le développement du site entre différents développeurs (indépendance des données, de l'affichage et des actions).

3. Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation décrit le comportement du système du point de vue utilisateur sous forme d'actions et de réactions. Il existe deux concepts fondamentaux dans la modélisation par les cas d'utilisation :

- Les acteurs qui agissent sur le système
- Les cas d'utilisations qui représentent les façons dont le système est manipulé par les acteurs.

Chaque cas d'utilisation indique une fonctionnalité du système déclenché par un acteur externe au système. Ce genre de diagramme permet de mettre en place et de comprendre les besoins des utilisateurs.

3.1. Identification des acteurs et leurs rôles

Au niveau de cette section, nous présentons les différents acteurs susceptibles d'interagir avec le système, mais tout d'abord, nous donnons une définition du concept acteur.

Acteur : le rôle joué par des entités externes qui interagissent directement avec le système étudié, il peut être un utilisateur, un matériel externe ou un autre système.

La mise en marche du système nécessite essentiellement trois acteurs :

- **Le responsable du stock (magasinier) :** c'est le gérant de l'application, son rôle consiste à gérer tout le système. En plus, il a la mission de la gestion des utilisateurs.
- **Le chef de département :** Il a le droit de visualiser les éléments du stock ainsi que les demandes effectuées de la part des enseignants appartenant à son département, il peut aussi faire une demande ou une réclamation.
- **Enseignant ou administratif :** il a le droit d'effectuer une demande ou une réclamation ainsi que visualiser ses demandes et suivre leurs états.

ACTEUR	ROLES
Responsable du stock	▪ Authentification. ▪ Gestion du stock • Lister les articles du stock. • Chercher un article. • Ajouter un article. • Modifier un article. • Supprimer un supprimer. ▪ Gestion des catégories • Lister les catégories. • Chercher une catégorie. • Ajouter une catégorie. • Modifier une catégorie. • Supprimer une catégorie. ▪ Gestion des demandes • Lister les demandes. • Vérifier les demandes (Accepter/Refuser). • Imprimer le reçu de décharge. ▪ Gestion des réclamations • Consulter les réclamations. • Vérifier les réclamations.

Tableau 1 : Les acteurs et leurs rôles

	▪ Gestion des départements • Lister les départements. • Ajouter un département. • Modifier un département. • Supprimer un département. ▪ Gestion des utilisateurs du système • Visualiser les utilisateurs. • Chercher un utilisateur. • Ajouter un utilisateur. • Modifier un utilisateur. • Supprimer un utilisateur. ▪ Consulter les statistiques
Chef de département	❖ Authentification. ❖ Demander un article. ❖ Faire une réclamation. ❖ Lister leurs demandes. ❖ Suivre leurs réclamations. ❖ Gestion des demandes/réclamations du département ❖ Consulter les demandes. ❖ Consulter les réclamations.
Enseignant	❖ Authentification. ❖ Demander un article. ❖ Faire une réclamation. ❖ Lister leurs demandes. ❖ Suivre leurs réclamations.

3.2. Les messages émis et reçus

Ce tableau va présenter les messages permettant de décrire les interactions de plus haut niveau entre les acteurs et le système. Tel que chaque acteur, envoi des messages qui déclenchent un comportement du système attendu par l'acteur dans le cadre de son activité.

Cas d'utilisation	Acteur	Messages émis/reçus
S'authentifier	Responsable du stock Chef de département Enseignant	**Emis :** Authentification et accès au système. **Reçu :** Demande d'authentification et connexion
Ajouter un article	Responsable du stock	**Emis :** Ajouter les informations d'un nouvel article. **Reçu :** Confirmation.
Modifier un article	Responsable du stock	**Emis :** Choisir l'article à modifier de la liste des articles. **Reçu :** Demande de spécification des changements et validation.
Supprimer un article	Responsable du stock	**Emis :** Choisir l'article à supprimer de la liste des articles. **Reçu :** Confirmation
Rechercher un article	Responsable du stock	**Emis :** Rechercher par nom/catégorie **Reçu :** Résultat de la recherche.
Ajouter une catégorie	Responsable du stock	**Emis :** Ajouter le nom de la nouvelle catégorie. **Reçu :** Confirmation.
Modifier une catégorie	Responsable du stock	**Emis :** Choisir la catégorie à modifier de la liste des catégories. **Reçu :** Confirmation.
Supprimer une catégorie	Responsable du stock	**Emis :** Choisir la catégorie à supprimer de la liste des catégories. **Reçu :** Confirmation.
Vérifier une demande	Responsable du stock	**Emis :** Choisir la demande à vérifier

		de la liste des demandes reçues, puis l'accepter ou la refuser. **Reçu :** Changement d'état de la demande.
Vérifier une réclamation.	Responsable du stock	**Emis :** Choisir la réclamation à vérifier de la liste des réclamations reçues, puis l'accepter ou la refuser. **Reçu :** Changement d'état de la réclamation.
Ajouter un département	Responsable du stock	**Emis :** Ajouter les informations du nouvel département. **Reçu :** Confirmation.
Modifier un département	Responsable du stock	**Emis :** Choisir le département à modifier de la liste des départements. **Reçu :** Spécifier les changements et valider.
Supprimer un département	Responsable du stock	**Emis :** Choisir le département à supprimer de la liste des départements. **Reçu :** Confirmation.
Ajouter un utilisateur (enseignant / chef de département).	Responsable du stock	**Emis :** Remplir les informations d'un nouveau personnel. **Reçu :** Confirmation.
Rechercher un utilisateur	Responsable du stock	**Emis :** Rechercher un utilisateur par nom / département **Reçu :** Résultat de la recherche.
Modifier un utilisateur	Responsable du stock	**Emis :** Choisir l'utilisateur à modifier de la liste des utilisateurs. **Reçu :** Spécifier les modifications et confirmer.
Supprimer un utilisateur	Responsable du stock	**Emis :** Choisir l'utilisateur à supprimer de la liste des utilisateurs. **Reçu :** Confirmation.
Effectuer une demande	Enseignant Chef de département	**Emis :** Choisir les articles désirés et remplir le formulaire de la demande.

		Reçu : Confirmation.
Faire une réclamation	Enseignant Chef de département	**Emis :** Choisir la demande sur laquelle il va faire la réclamation et remplir le formulaire. **Reçu :** Confirmation.
Consulter les demandes de son département.	Chef de département	**Emis :** Choisir la demande à visualiser de la liste des demandes. **Reçu :** Affichage de la demande choisie.
Consulter les réclamations de son département.	Chef de département	**Emis :** Choisir la réclamation à visualiser de la liste des réclamations. **Reçu :** Affichage de la réclamation choisie.

Tableau 2 : Les messages émis et reçus

3.3. Diagramme de cas d'utilisation relatif au responsable du stock

Ces cas d'utilisation englobent l'ensemble des opérations majeurs effectuées par le responsable du stock.

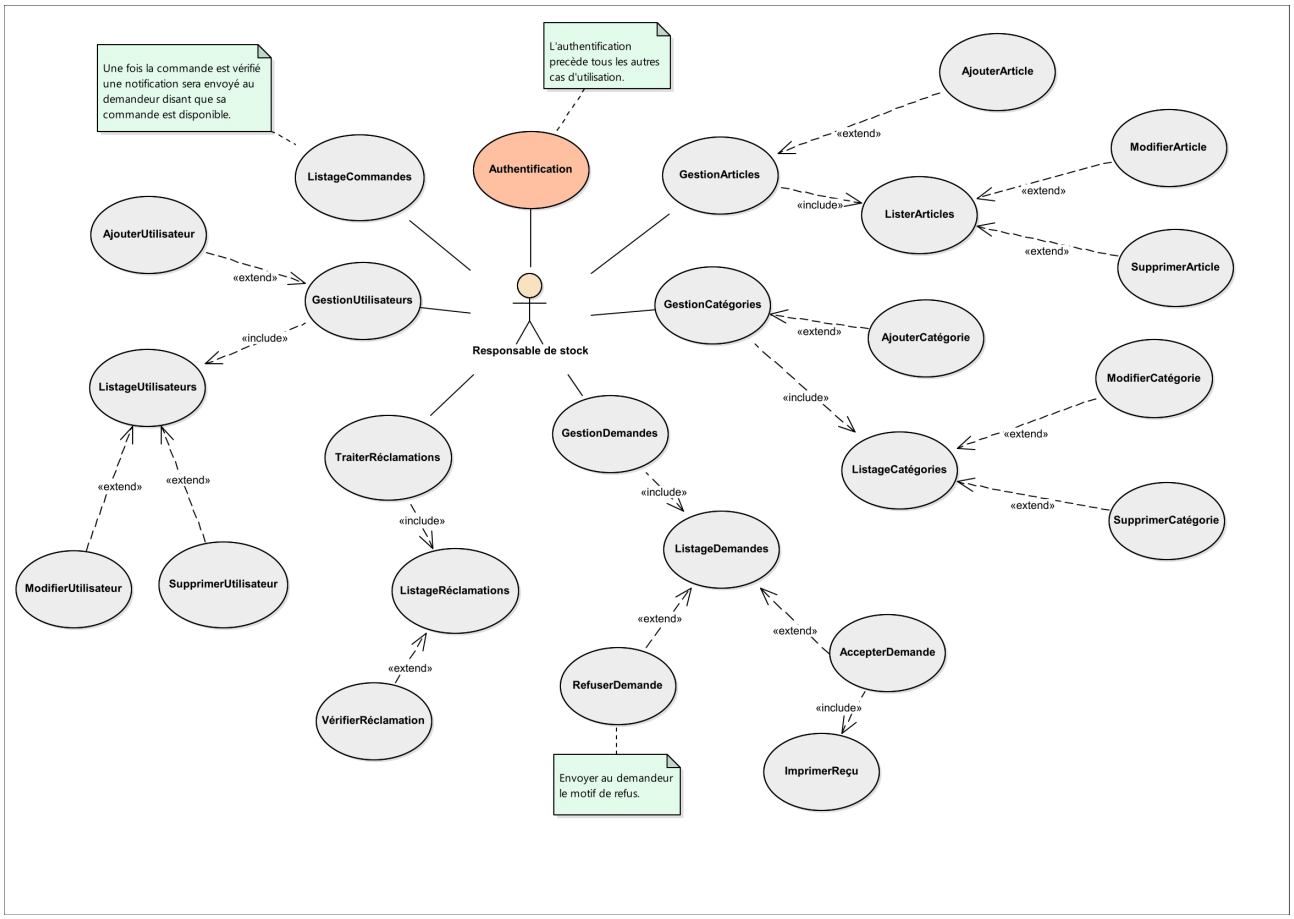

Figure 5 : Diagramme de cas d'utilisation relatif au responsable du stock

3.4. Diagramme de cas d'utilisation relatif aux enseignants

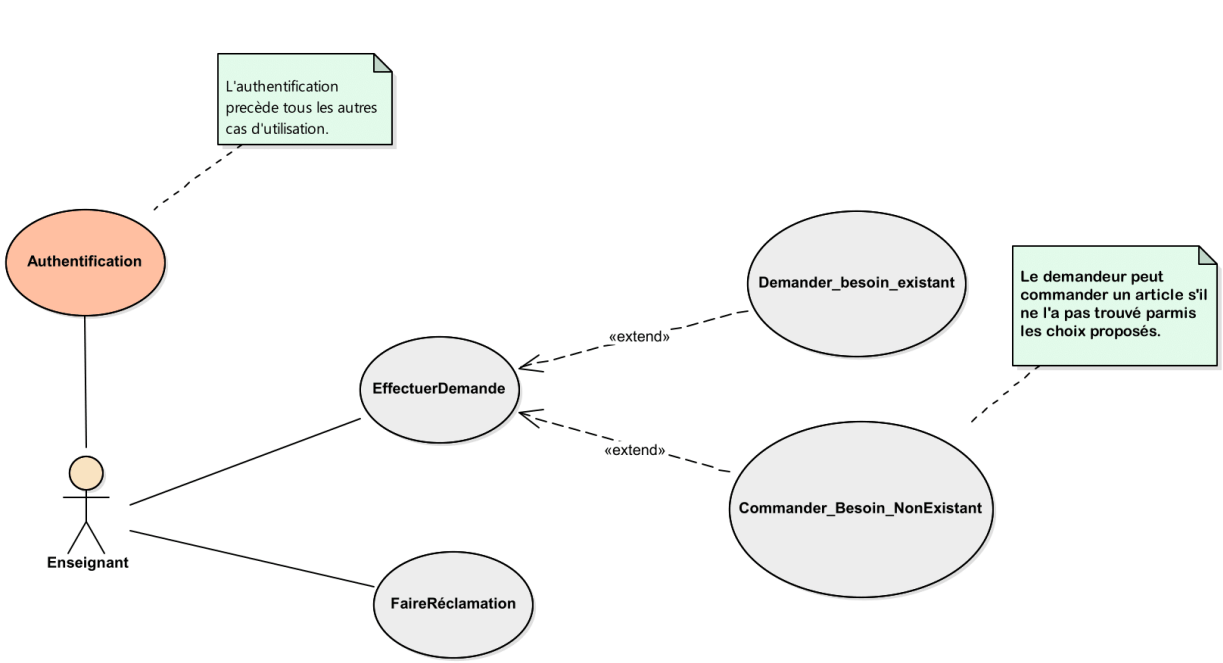

Figure 6 : Diagramme de cas d'utilisation relatif à l'enseignant

3.5. Diagramme de cas d'utilisation relatif au chef de département

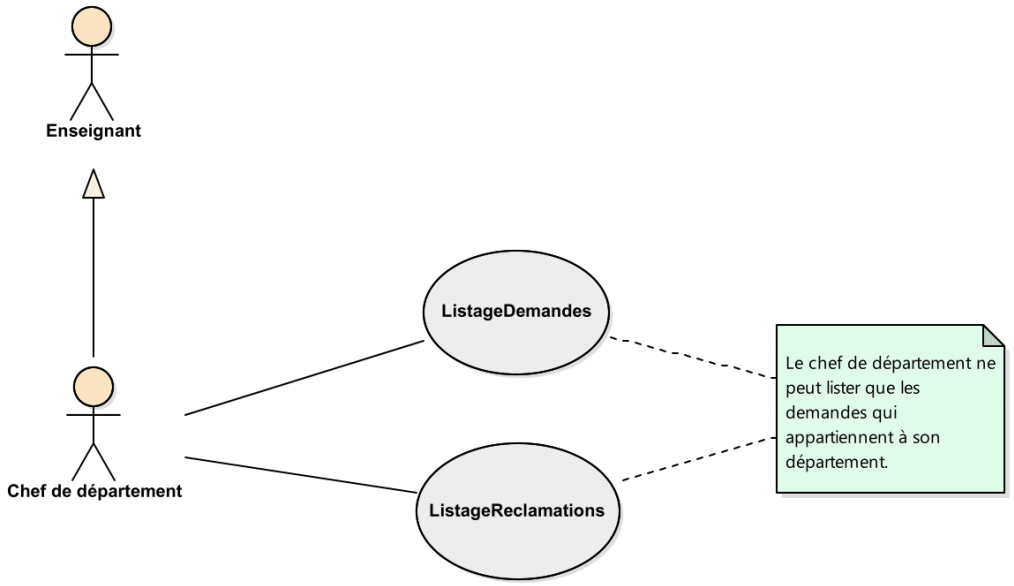

Figure 7 : Diagramme de cas d'utilisation relatif au chef de département

4. Diagrammes de séquence

Parmi les diagrammes intéressants d'UML, on trouve le diagramme de cas de séquence qui illustre une représentation graphique des interactions entre l'acteur et le système selon un ordre chronologique dans la formulation Unified Modeling Language.

L'utilité du diagramme de séquence est de montrer les interactions d'objet dans le cadre d'un scénario de cas d'utilisation.

La dimension verticale du diagramme représente le temps, permettant de visualiser l'enchaînement des actions dans le temps, et de spécifier la naissance et la mort d'objets. Les périodes d'activité des objets sont symbolisées par des rectangles, et ces objets dialoguent à l'aide de messages.

Donc, vu le service offert par les diagrammes de séquence, cette partie est consacrée pour citer ceux des cas d'utilisation les plus importants dans l'application, ce qui permet de mieux voir et tracer l'enchainement du projet.

- **Authentification**

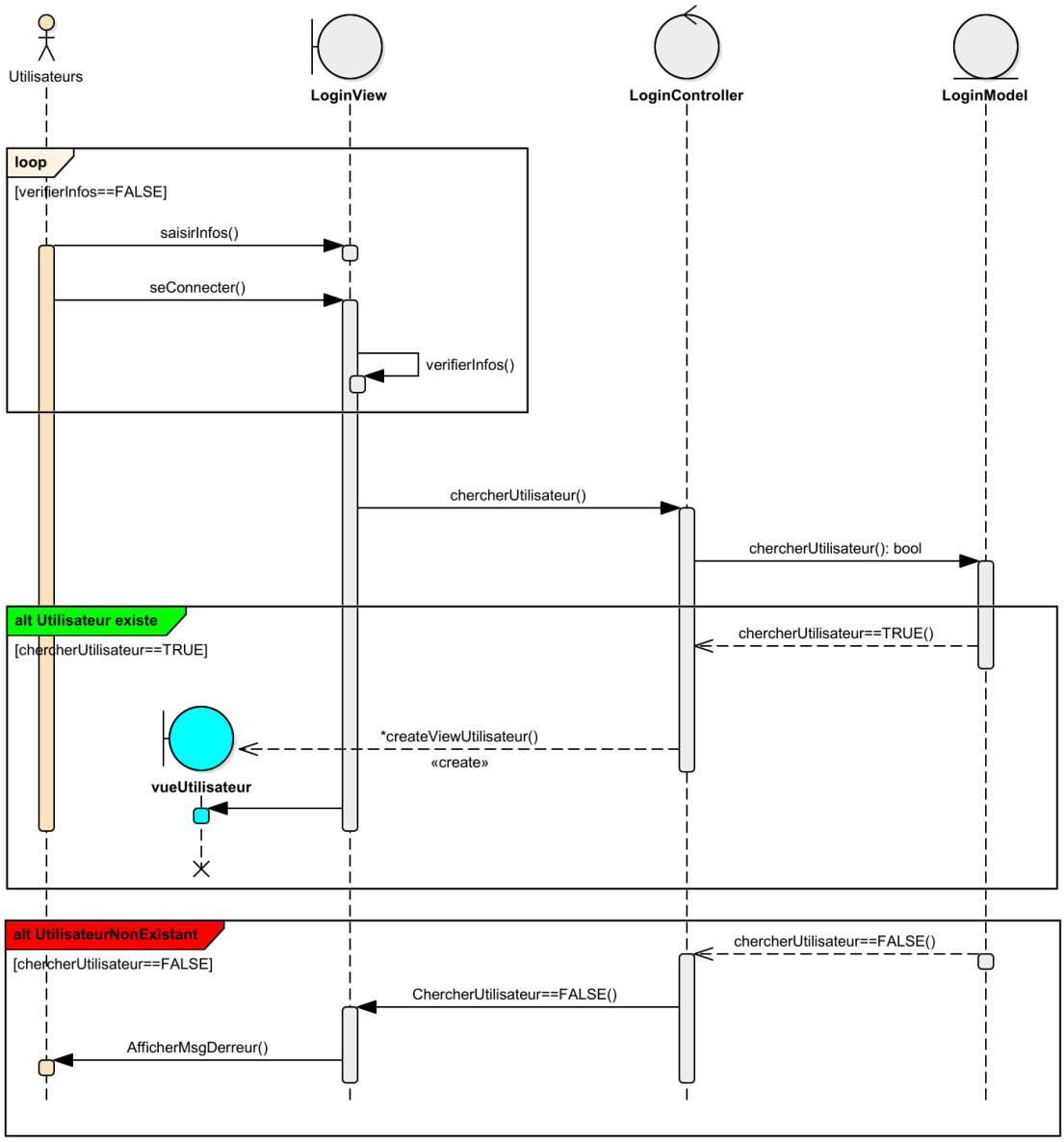

Figure 8 : Diagramme de séquence - Authentification -

- **Ajouter article**

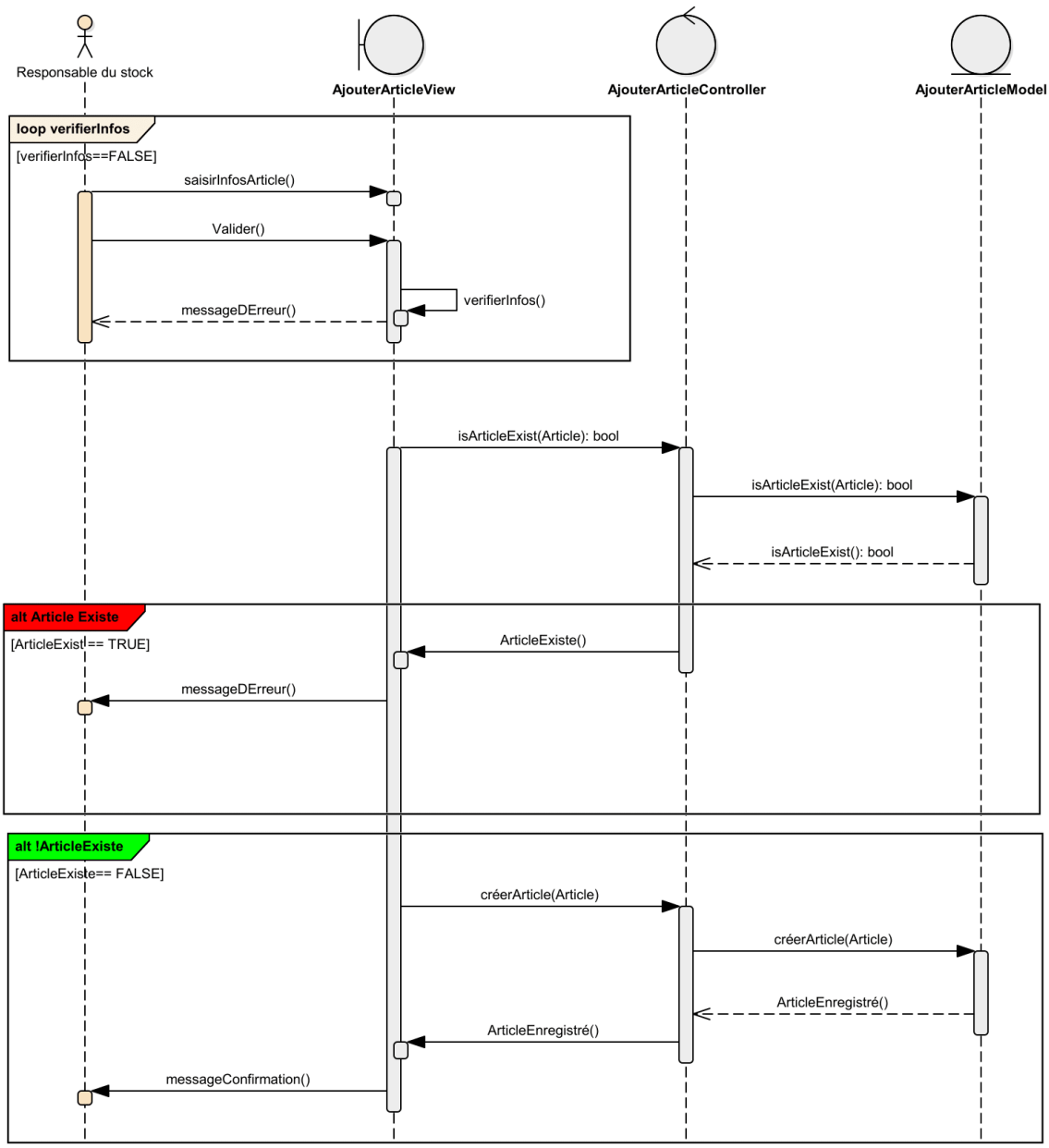

Figure 9 : Diagramme de séquence - Ajouter Article -

- **Effectuer demande**

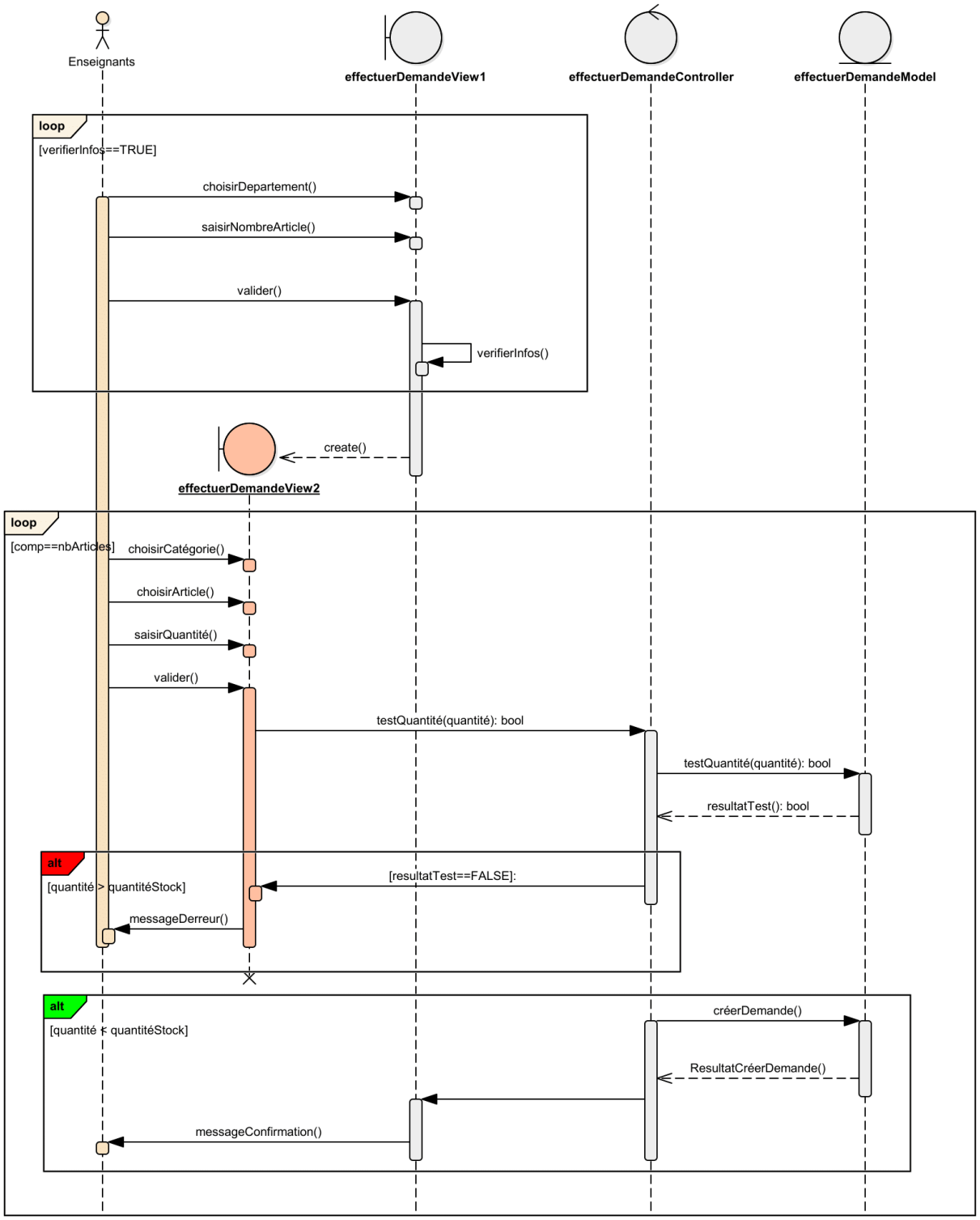

Figure 10 : Diagramme de séquence - Effectuer demande -

- **Gestion demandes**

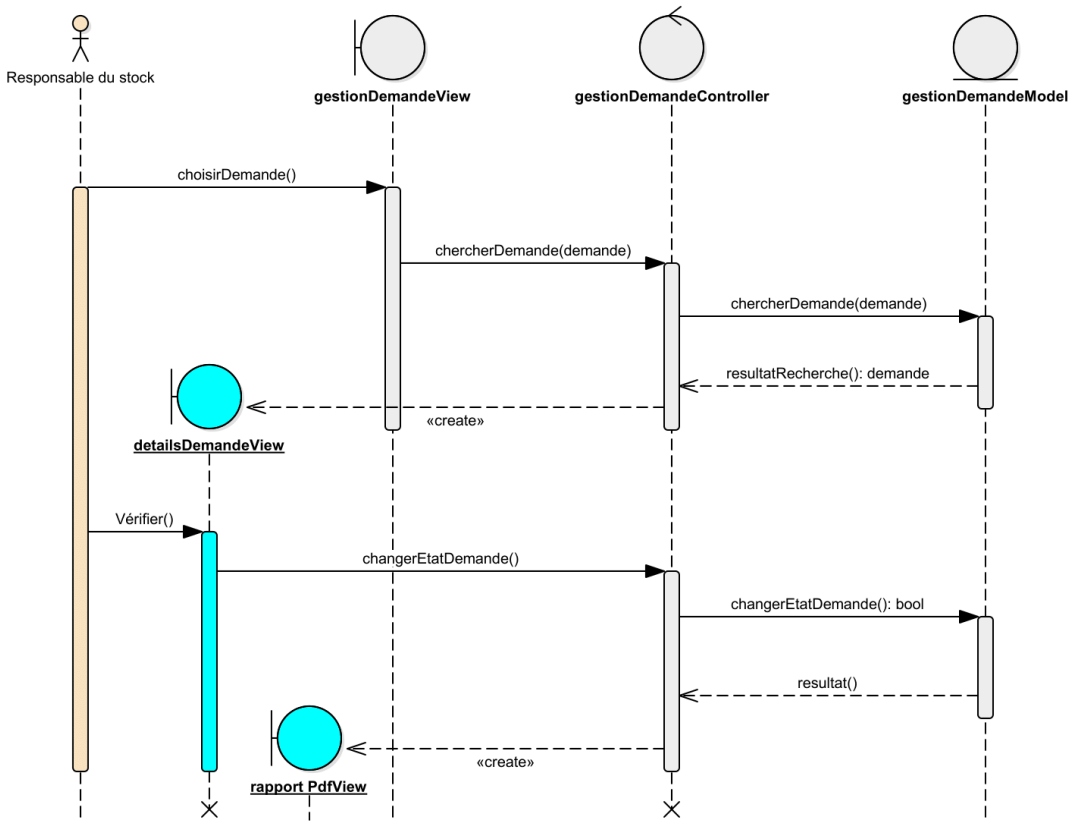

Figure 11 : Diagramme de séquence - Gestion demandes -

- **Gestion réclamations**

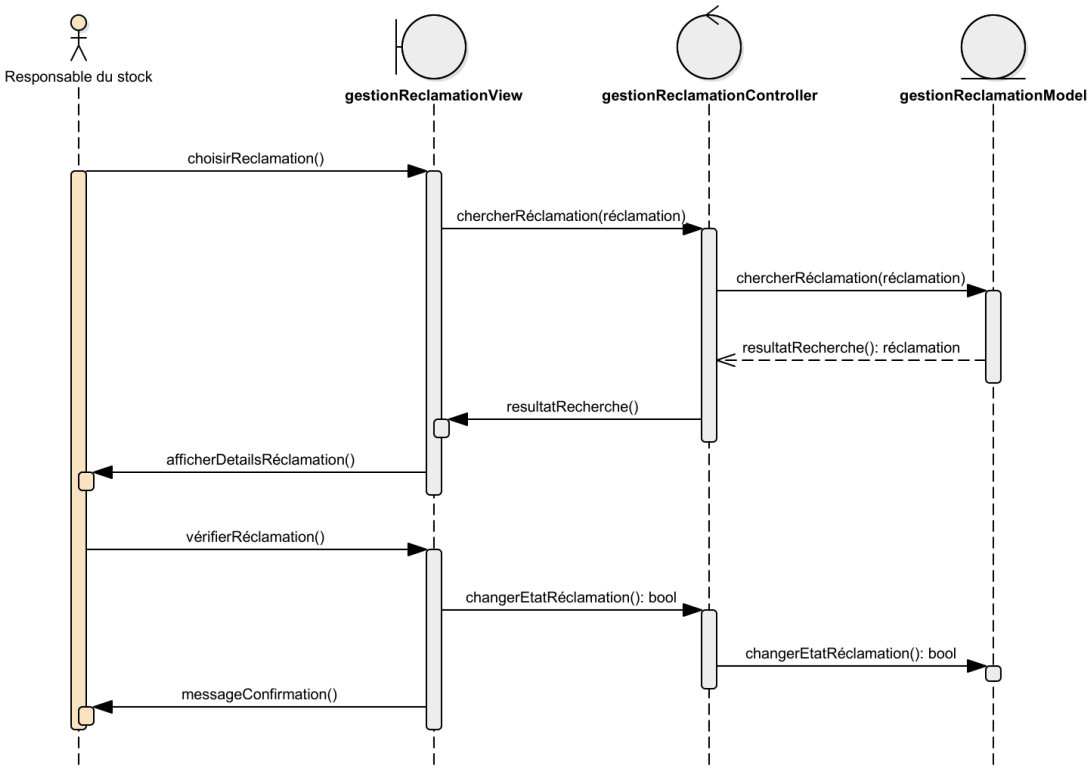

Figure 12 : Diagramme de séquence - Gestion réclamations -

5. Diagramme de classes

Le diagramme de classes exprime la structure statique du système en termes de classes et de relations entre ces classes. L'intérêt du diagramme de classes est de modéliser les entités du système d'information. Ces informations sont regroupées ensuite dans des classes.

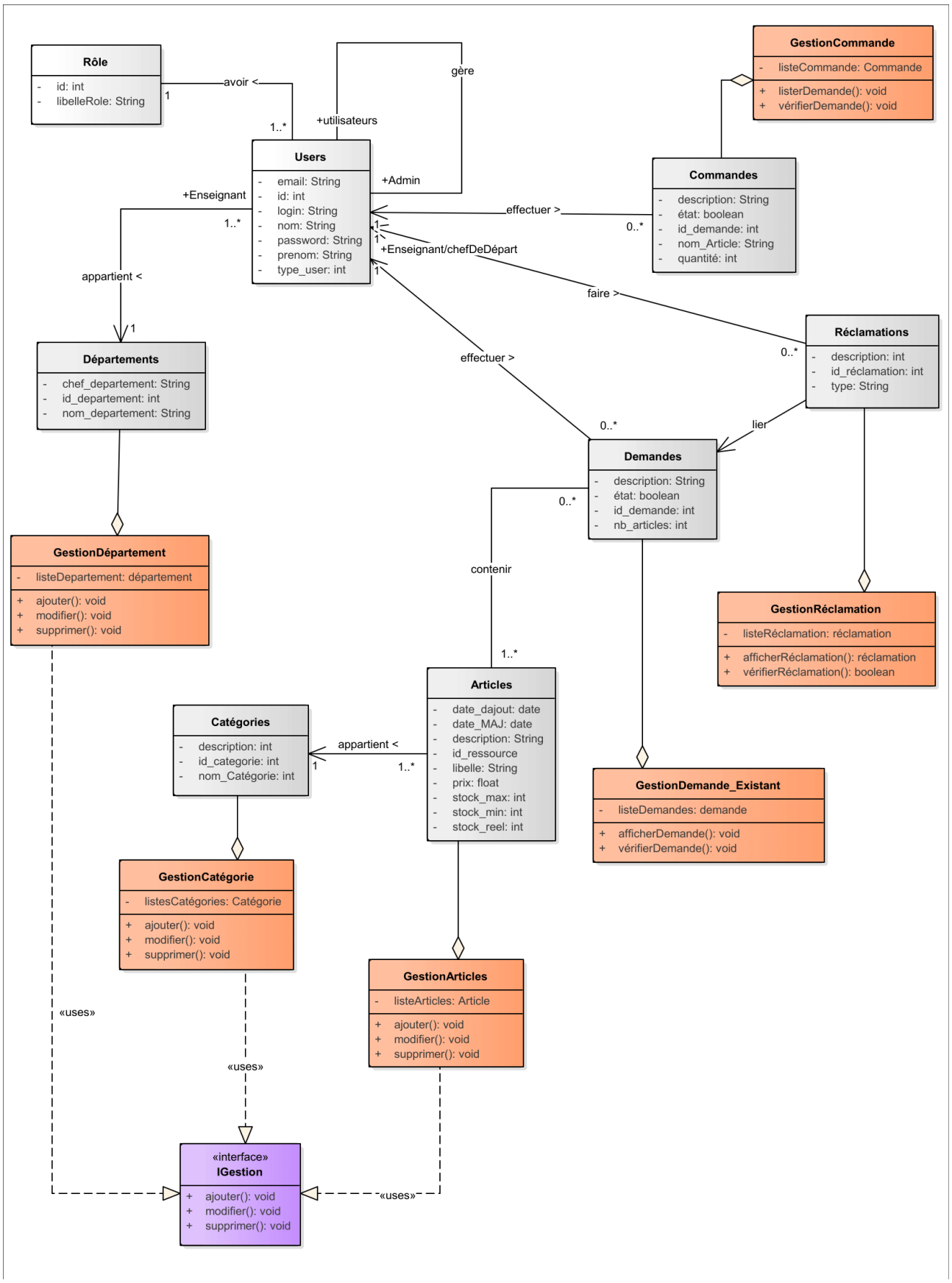

Figure 13 : Diagramme de classes

6. Modèle logique de données

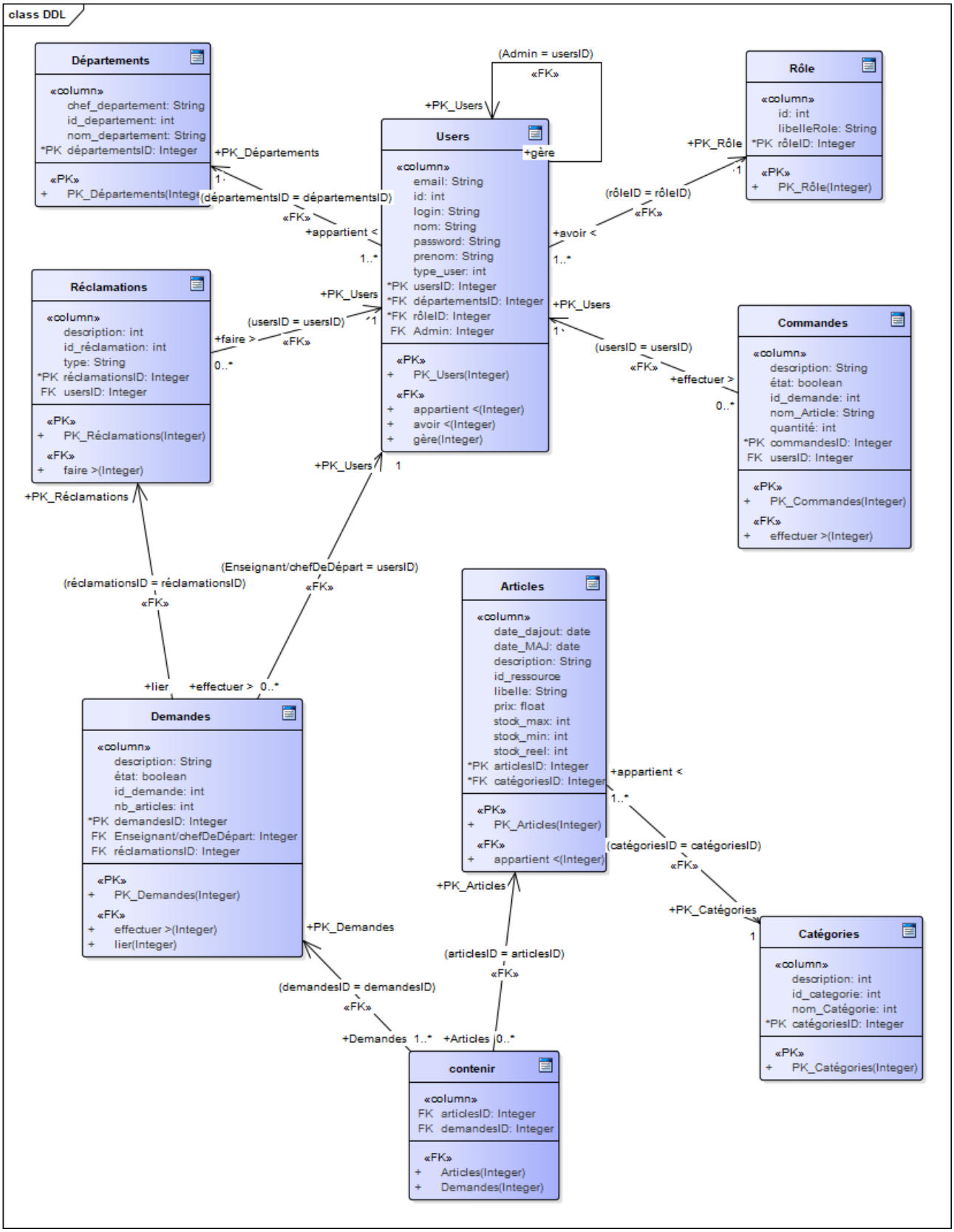

Figure 14 : Modèle logique de données

Chapitre III

Mise en œuvre du projet

1. Outils de développement

1.1. Technologies Web

- ***HTML***

L'HTML est un langage informatique utilisé sur Internet. Ce langage est utilisé pour créer des pages web. L'acronyme signifie *HyperText Markup Language*, ce langage permet de réaliser de l'hypertexte à base d'une structure de balisage. L'HTML5 est le successeur de l'HTML 4.01, ça veut dire qu'il s'agit toujours du HTML à la différence de quelques nouvelles balises.

De plus, la version 5 est aujourd'hui compatible avec la majorité des navigateurs et répond aux normes W3C (C'est une communauté internationale où les membres, une équipe à plein temps, et le public travaillent ensemble pour développer les standards du web).

Fondamentalement HTML5 a ses nombreuses nouvelles fonctionnalités syntaxiques, qui comprennent le <video>, <audio>, et des éléments de <canvas>. En raison de ces nouveaux éléments, il sera très facile d'intégrer du contenu multimédia et graphique pour le Web sans utiliser le flash et plugins tiers. Il y'a aussi des nouveaux éléments comme <section>, <article>, <header> et <nav> qui enrichissent la valeur sémantique du document.

- ***CSS***

Les feuilles de styles (en anglais "Cascading Style Sheets", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Les styles permettent de définir des règles appliquées à un ou plusieurs documents HTML. Ces règles portent sur le positionnement des éléments, l'alignement, les polices de caractères, les couleurs, les marges et espacements, les bordures, les images de fond, etc.

- ***PHP***

PHP est un langage de script coté serveur qui a été conçu spécifiquement pour le Web. Le code PHP est inclus dans une page HTML et sera exécuté à chaque fois qu'un visiteur affichera la page. Le code PHP est interprété au niveau du serveur web et génère du code HTML ou toute autre donnée affichable dans le navigateur de l'utilisateur. PHP a été conçu en 1994 par Rasmus Lerdorf. Il a ensuite été adopté par d'autres personnes talentueuses et réécrit quatre fois avant de devenir le produit abouti que nous connaissons aujourd'hui.

En novembre 2007, il était installé sur plus de 21 millions de domaines et sa croissance est rapide. PHP est un projet open-source, ce qui signifie que vous pouvez vous procurer son code, l'utiliser, le modifier et le redistribuer gratuitement. La dernière version principale de

PHP est la version 5. Elle bénéficie d'une réécriture complète du moteur Zend et de quelques améliorations importantes au niveau du langage.

- ***JavaScript***

Le JavaScript est un langage de script incorporé dans un document HTML. Historiquement il s'agit même du premier langage de script pour le Web. Ce langage est un langage de programmation qui permet d'apporter des améliorations au langage HTML en permettant d'exécuter des commandes du côté client, c'est-à-dire au niveau du navigateur et non du serveur web.

- ***JQuery***

JQuery est une bibliothèque JavaScript libre qui porte sur l'interaction entre JavaScript (comprenant Ajax) et HTML, et a pour but de simplifier des commandes communes de

JavaScript.

- ***MySQL***

MySQL est un système de gestion de base de données relationnelle (SGBDR). Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde.

1.2. Les Framework de développement web

- ***Présentation***

Un Framework ou structure logicielle est un ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel (architecture).

- ***Avantages***

Le premier avantage est la liberté de créer les composantes nécessaires à l'application de la façon la plus conviviale. Le deuxième est la rapidité car pas mal de composantes sont prédéfinis est

disponible et prêtes à être intégrées. Le troisième est la qualité des projets faits avec ces Framework. Le quatrième est la cohérence. Etc.

- ***Exemples de Framework web PHP***

Laravel – Slim – Symfony - Zend Framework - CodeIgniter - FuelPHP - Yii - CakePHP - QCodo - Jelix - Agavi - Atomik - Copix - Kumbia PHP - PRADO - WebSite-PHP – Phalcon

1.3. Choix technologiques

- ***Outils de programmation choisis***

Le langage le plus approprié pour la programmation des sites web dynamiques, en plus de HTML qui sert de base, est le langage PHP. Ce dernier peut être combiné au système de gestion de base de données MySQL. C'est un langage coté serveur qui s'intègre dans les pages HTML. Il permet, entre autres, de rendre automatiques des tâches répétitives, notamment grâce à l'utilisation d'objets qui sont en quelques sortes des petits programmes permettant d'accomplir les tâches qui lui sont confiées grâce à l'exécution de ses méthodes.

De plus, ce langage permet de manipuler les sessions. Une session correspond à l'intervalle de temps compris entre le premier accès par un utilisateur au site web et la clôture de toutes les fenêtres de son navigateur. Les sessions permettront de stocker les informations concernant le membre connecté, PHP gère très bien les sessions et elles sont plus simples à utiliser et à mettre en œuvre. Pour la mise en forme des pages web nous avons choisi l'outil CSS.

L'outil Bootstrap a été choisi aussi. C'est un outil web qui facilite la création de sites internet et d'applications web. Il contient des modèles HTML et CSS qui permettent de créer rapidement des formulaires, des boutons, des outils de navigation et d'autres éléments dynamiques.

- ***Design pattern choisi***

Le pattern MVC

Le pattern MVC est un modèle d'architecture trois tiers, qui propose de séparer les données (modèle), les traitements (contrôleur) et la présentation (vue), pour pouvoir ultérieurement modifier une couche sans avoir à toucher aux autres. C'est un concept très puissant qui intervient dans la réalisation des applications. Le principe d'une telle structure est de diviser l'application en 3 parties distinctes :

- Les modèles : ils représentent les données de l'application et permettent l'interaction avec la base de données.

- Les vues : ce sont les représentations (les Template) des résultats de la requête que l'utilisateur effectue à chaque fois.
- Les contrôleurs : ils interceptent toutes les requêtes faites par les utilisateurs.

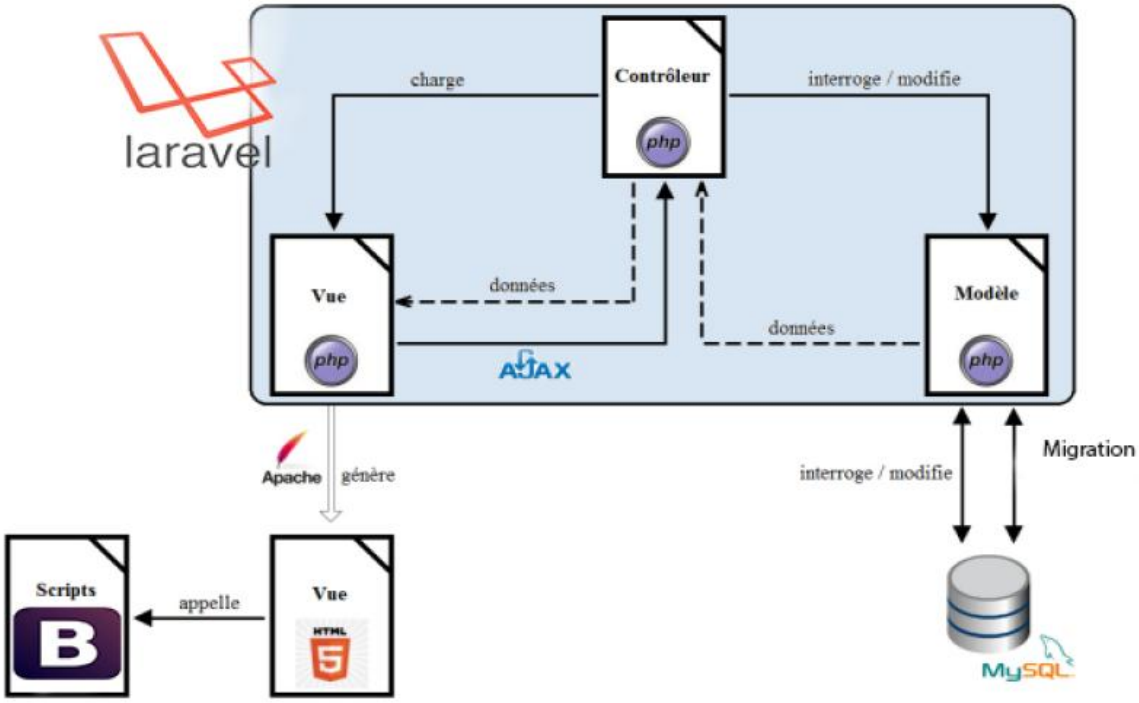

Figure 15 : schéma d'exécution d'un script avec le pattern MVC de Laravel

- ***Framework choisi***

Dans cette section nous présenterons les critères utilisés pour choisir le Framework Laravel.

Dans un premier temps nous avons fait une petite recherche sur les tendances en termes de popularité des Framework PHP courantes. La figure 14 montre les statistiques concernant les 15 meilleurs Framework PHP gratuits. Cette même figure relève que le Framework Laravel est le plus populaire parmi ces 15 Framework.

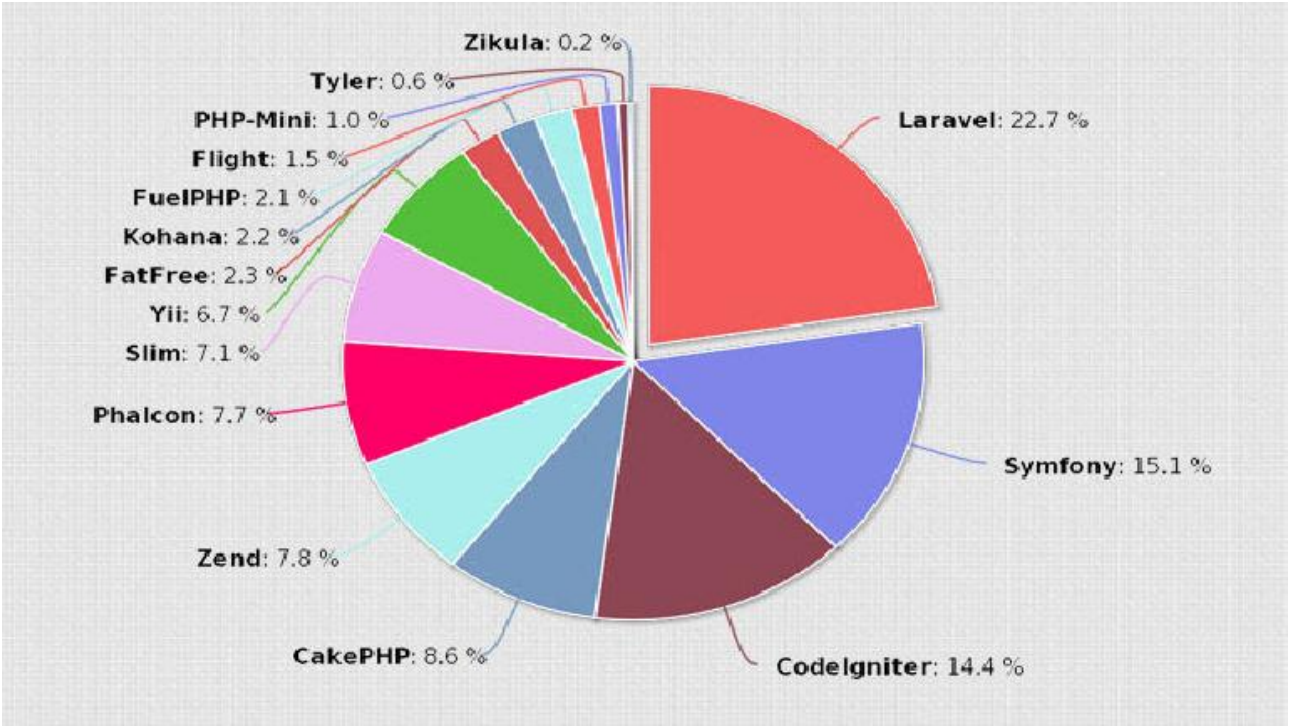

Figure 16 : statistiques des 15 Meilleurs Framework PHP gratuits

Laravel est un Framework d'application web PHP open-source, destiné pour le développement d'applications. Ses principales fonctionnalités sont :

- Composer en tant que gestionnaire de dépendance pour ajouter des packages de structure différentes et spécifiques disponible à partir du référentiel Packagist permettant ainsi le support de la majorité des technologies actuelle.
- Eloquent ORM (mapping objet-relationnel) est une implémentation de PHP avancé du motif d'enregistrement actif, en fournissant en même temps des méthodes internes pour appliquer des contraintes sur les relations entre les objets de base de données.
- Laravel dispose d'un moteur de Template "Blade" qui combine un ou plusieurs modèles avec un modèle de données pour produire des nouvelles vues, en traduisant les modèles dans le code PHP en cache pour améliorer les performances.
- Migrations : fournissent un système de contrôle de version pour les schémas de base de données, ce qui permet d'associer des changements dans le code source de l'application et les modifications nécessaires à la mise en page de base de données. En

conséquence, le déploiement et la mise à jour des applications est considérablement simplifié.

Nous trouvons dans Laravel :

- Un système de routage perfectionné (RESTFUL et ressources),
- Un créateur de requêtes SQL et un ORM performants,
- Un moteur de Template efficace,
- Un système d'authentification pour les connexions,
- Un système de validation,
- Un système de pagination,
- Un système de migration pour les bases de données,
- Un système d'envoi d'emails,
- Un système de cache,
- Une gestion des sessions...

Le pattern implémenté par Laravel est le pattern MVC (Modèle-Vue-Contrôleur). En plus, Laravel est fondamentalement orienté objet. La POO est un paradigme qui s'éloigne radicalement de la programmation procédurale. Avec la POO tout le code est placé dans des classes qui découlent d'interfaces qui établissent des contrats de fonctionnement. Avec la POO on manipule des objets.

En conclusion, Laravel est un Framework très complet et très puissant. Il a la particularité d'être plus permissive et plus souple que son confrère Symfony. Le fait qu'il soit très facile d'accès avec sa documentation complète et sa communauté active rend Laravel très attractif lors du choix d'un Framework.

2. Présentation de l'application

Dans cette partie nous présenterons les interfaces les plus importantes de notre application regroupée par acteur.

Tous les utilisateurs de l'application disposent d'un compte utilisateur et peuvent se connecter à distance à l'aide d'un nom d'utilisateur et un mot de passe. En plus, le profil de chaque utilisateur lui permet d'effectuer les tâches qui lui sont propre, et qui sont déjà présentées dans le chapitre II.

- ***Fenêtre d'authentification***

Figure 17 : Fenêtre d'authentification

- Message d'erreur -Utilisateur introuvable-

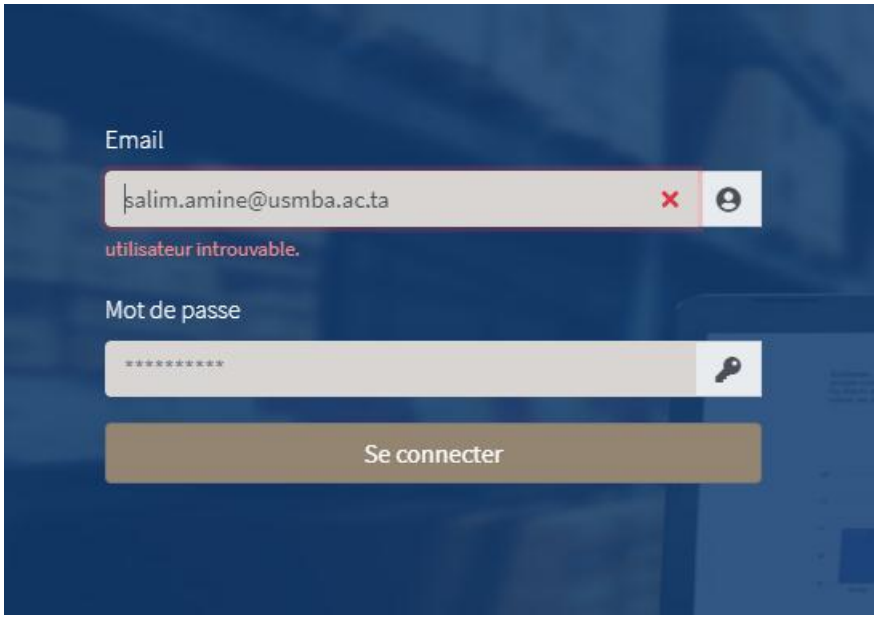

Figure 18 : Message d'erreur - utilisateur introuvable -

2.1. Interfaces du responsable du stock

- ***Accueil***

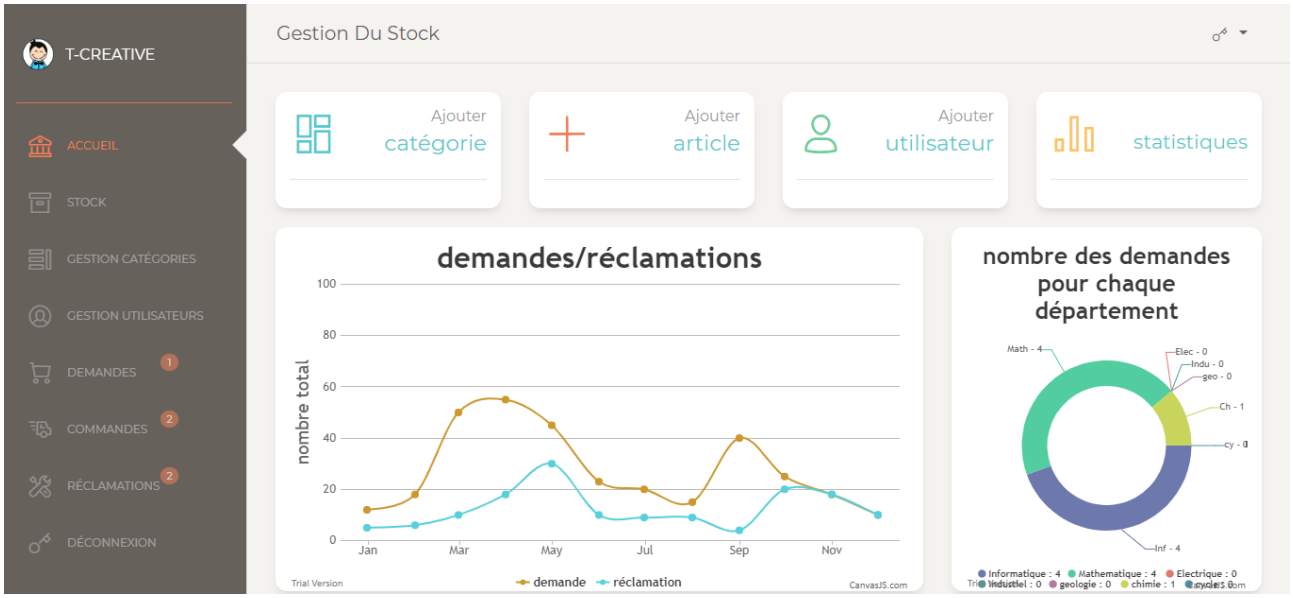

Figure 19 : Fenêtre d'accueil du responsable de stock

- ***Fenêtre de gestion du stock***
  - Liste des articles

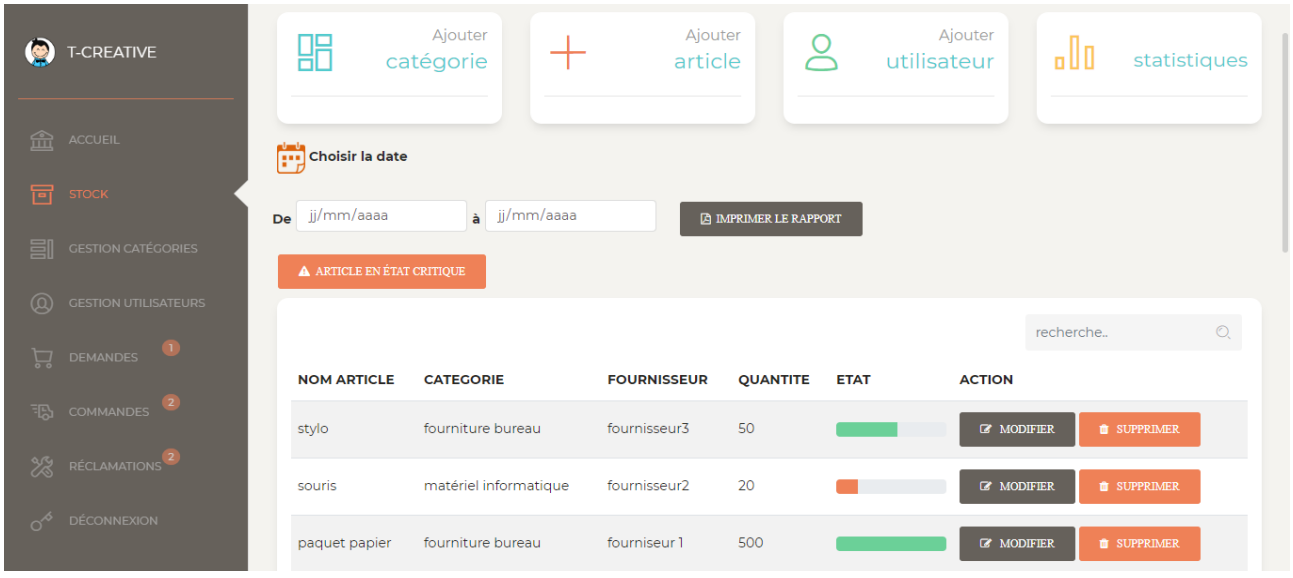

Figure 20 : Fenêtre de listes des articles

- Ajouter un article

Figure 21 : Formulaire d'ajouter un article

- Message de confirmation

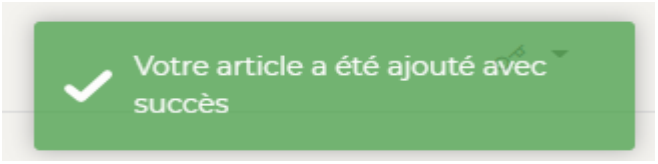

Figure 22 : Message de confirmation -Ajouter article –

- Les articles en état critique

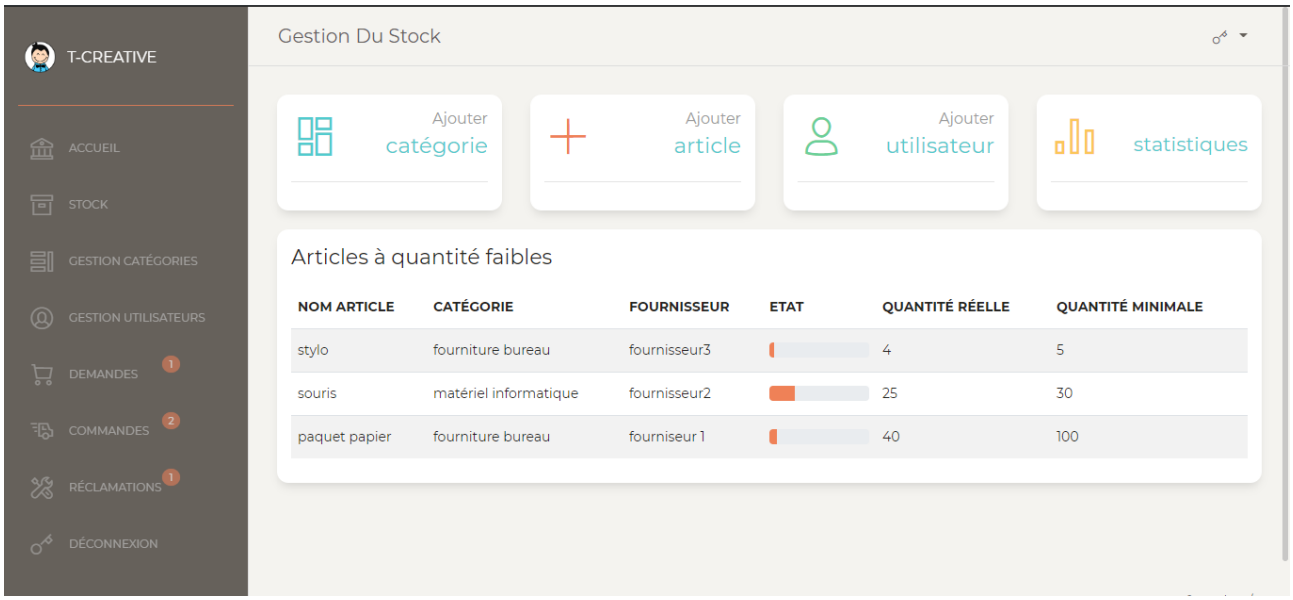

Figure 23 : Les articles en état critique

- Rapport

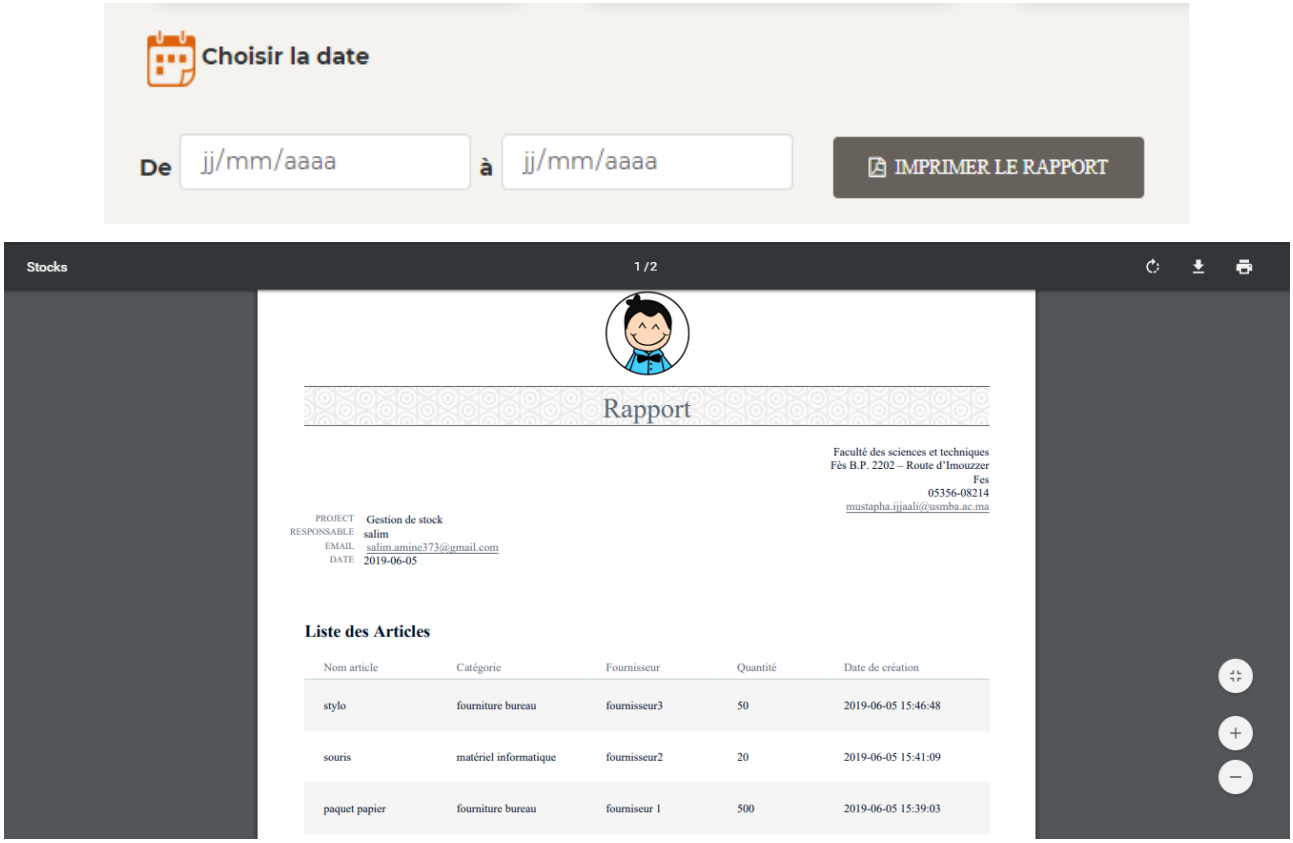

Figure 24 : Imprimer rapport PDF

- ***Fenêtre de gestion des utilisateurs***

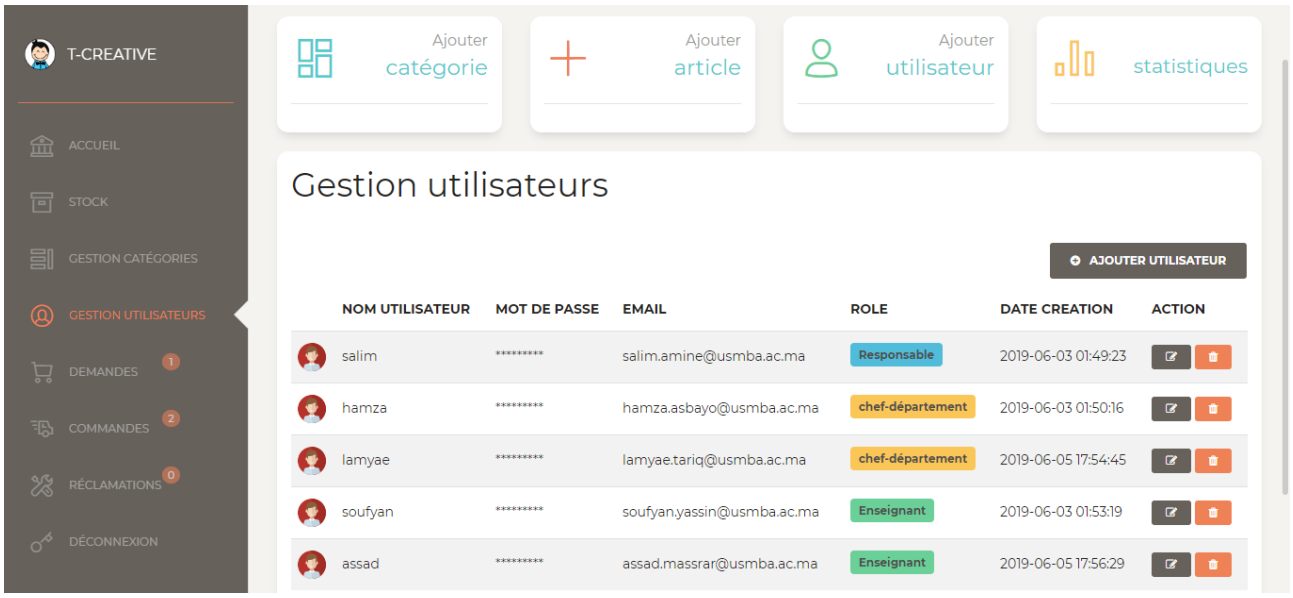

Figure 25 : Fenêtre de gestion des utilisateurs

- Ajouter utilisateur

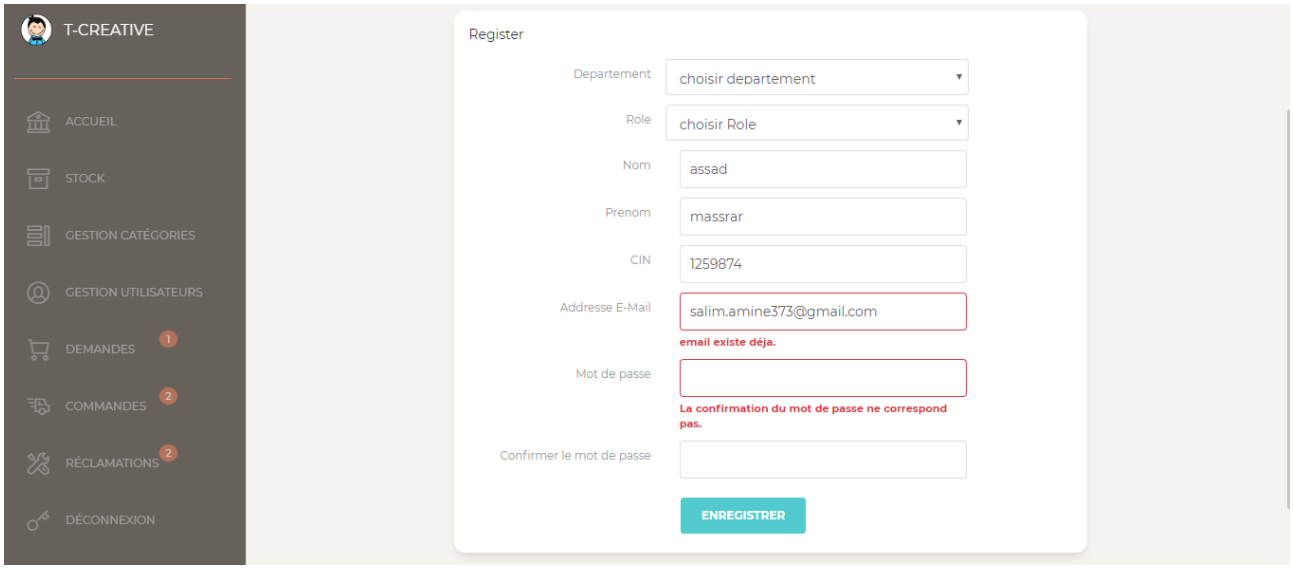

Figure 26 : Ajouter un nouvel utilisateur

- ***Fenêtre de gestion des demandes***

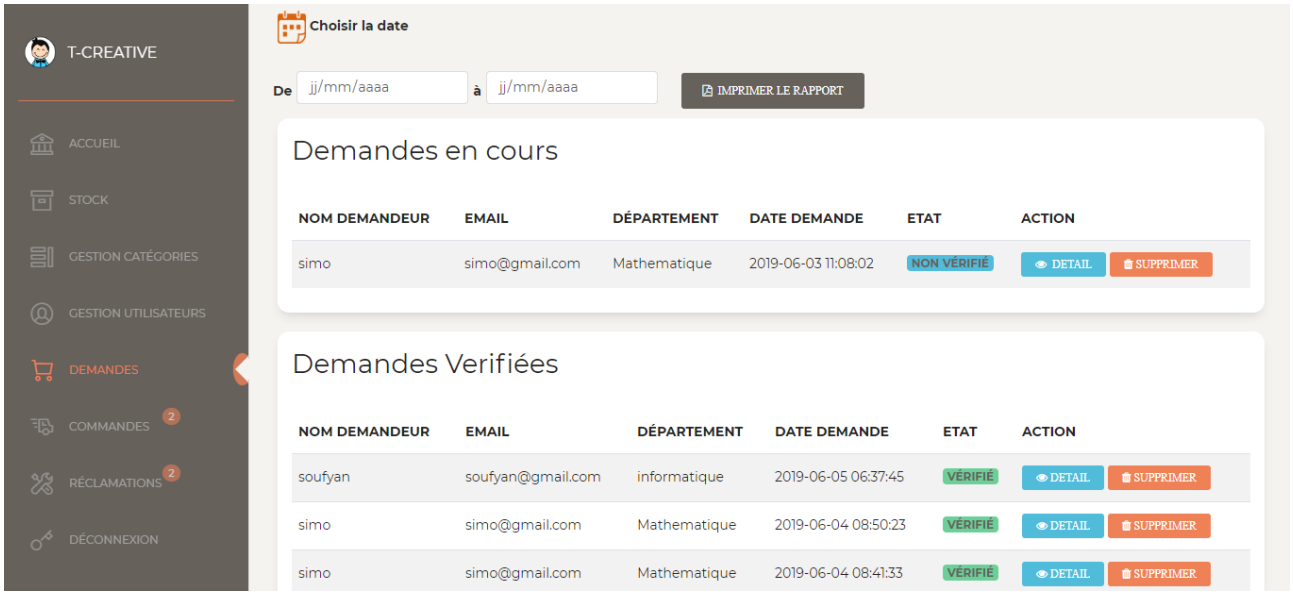

Figure 27 : Fenêtre de gestion des demandes

- Détails d'une demande

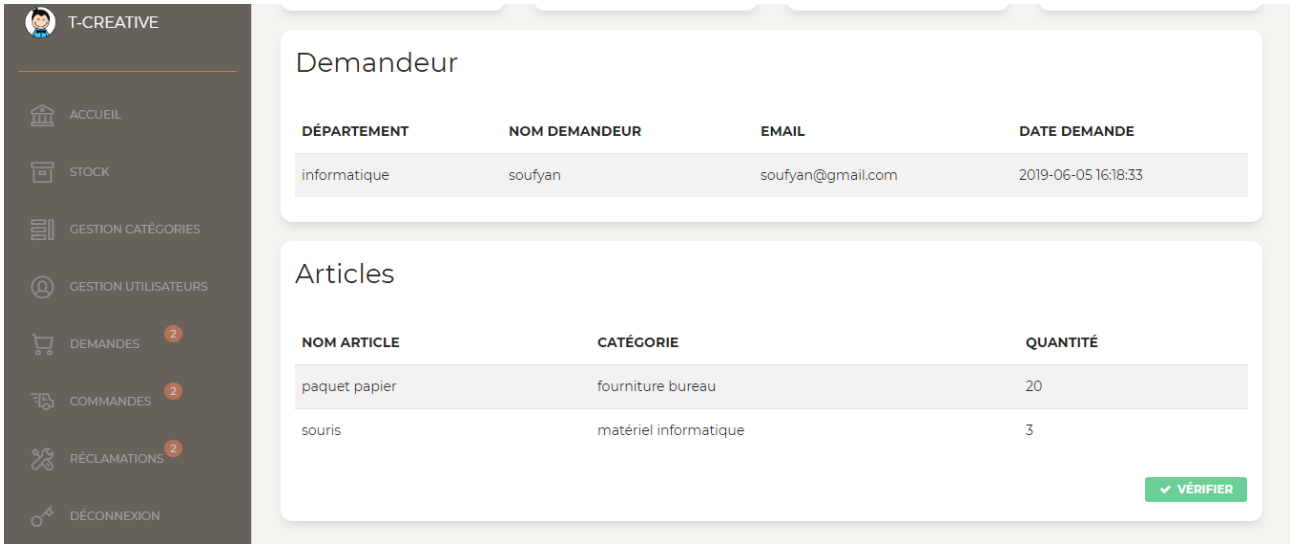

Figure 28 : Afficher les détails d'une demande

- Message pour confirmer la suppression d'une demande

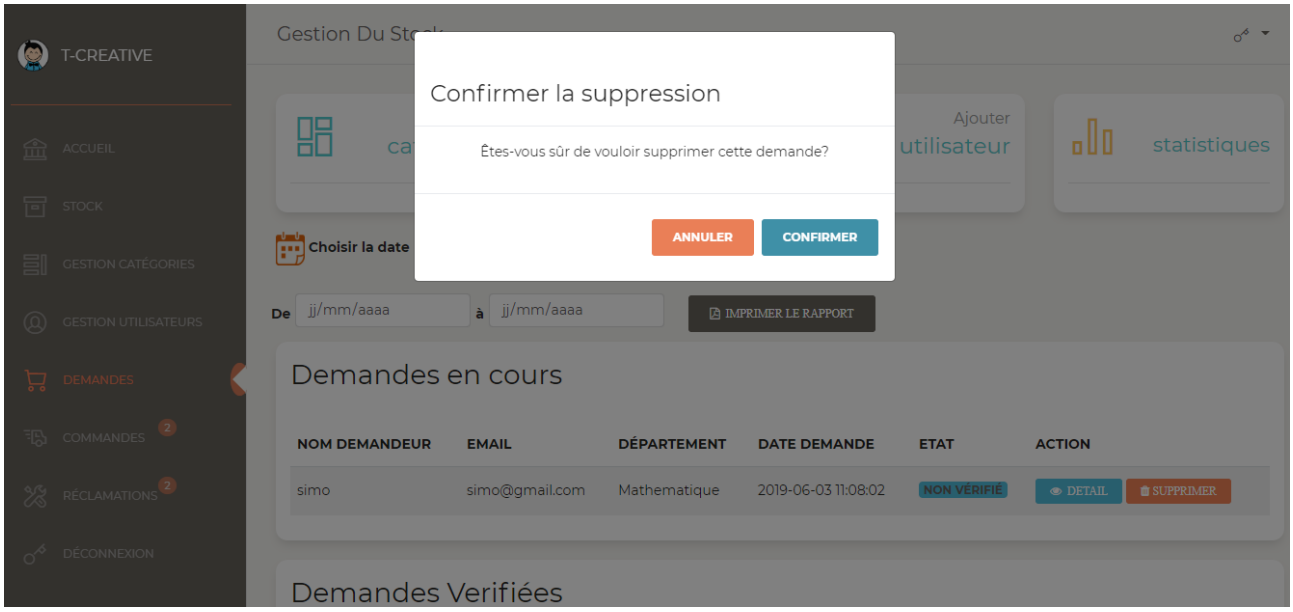

Figure 29 : Confirmer la suppression

- ***Fenêtre de gestion des réclamations***

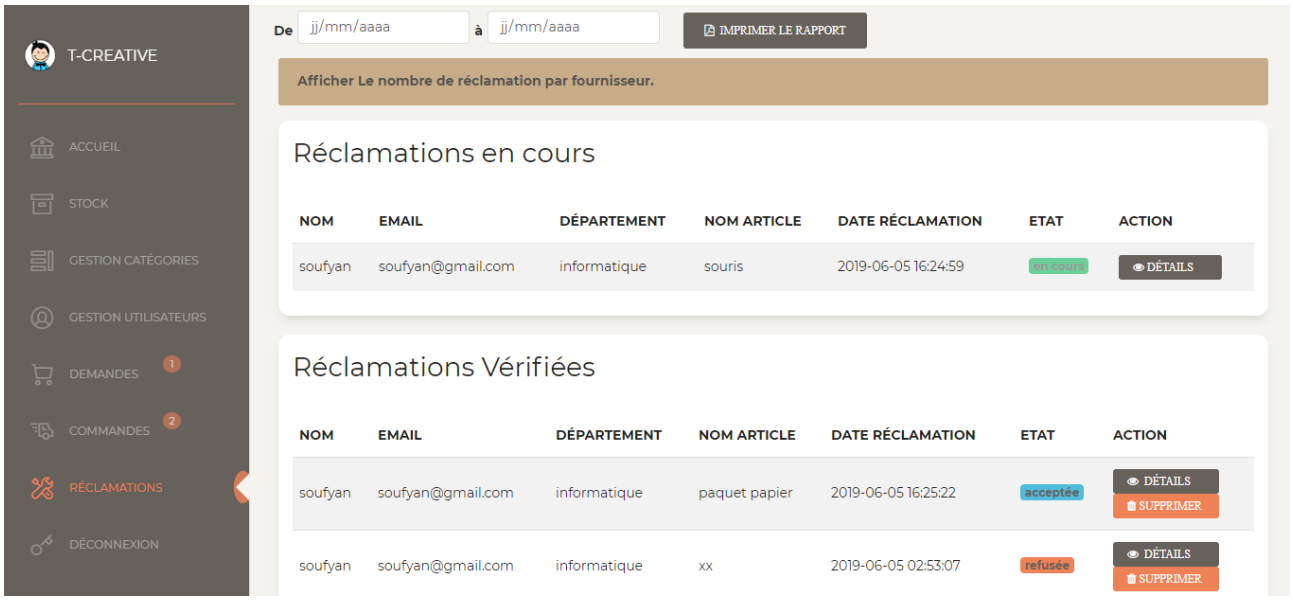

Figure 30 : Fenêtre de gestion des réclamations

- Détails d'une réclamation

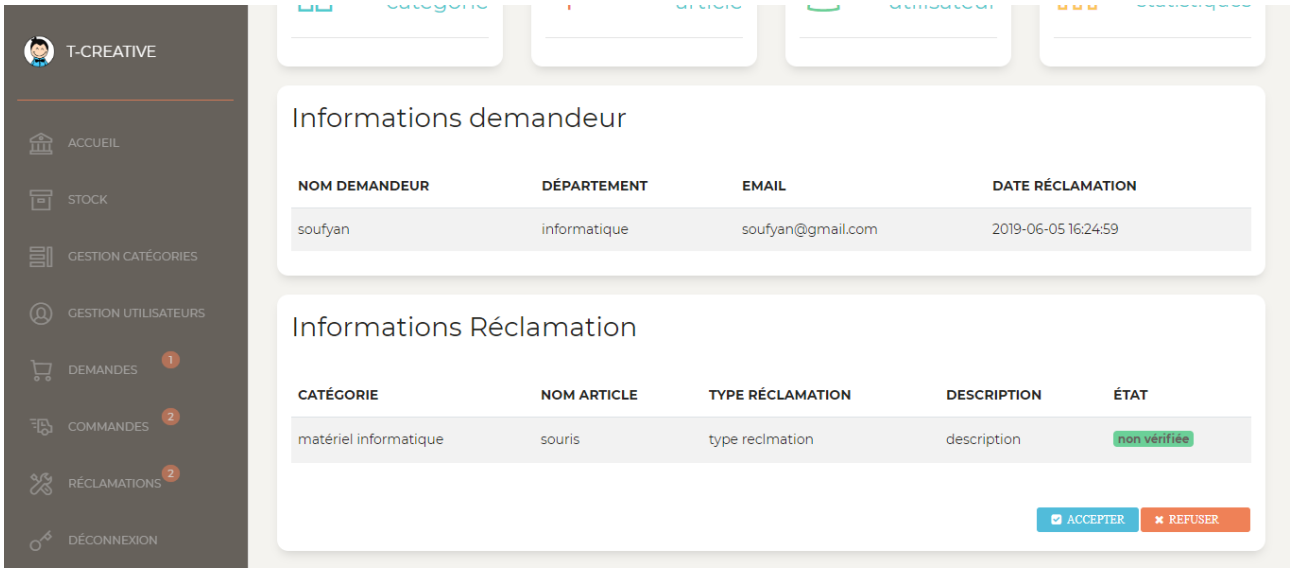

Figure 31 : Afficher les détails d'une réclamation

2.2. Interfaces des enseignants

- Demander Article

Figure 32 : Effectuer une demande - interface1 –

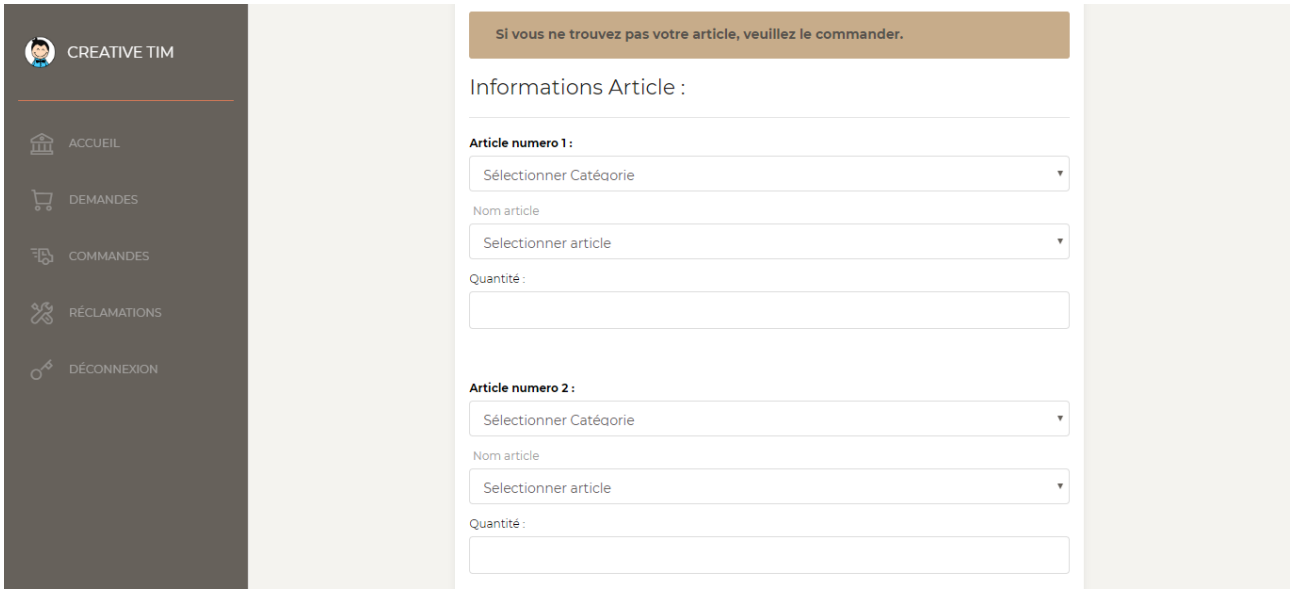

Figure 33 : Effectuer une demande – interface2 -

- Message d'erreur si la quantité demandée n'est pas disponible

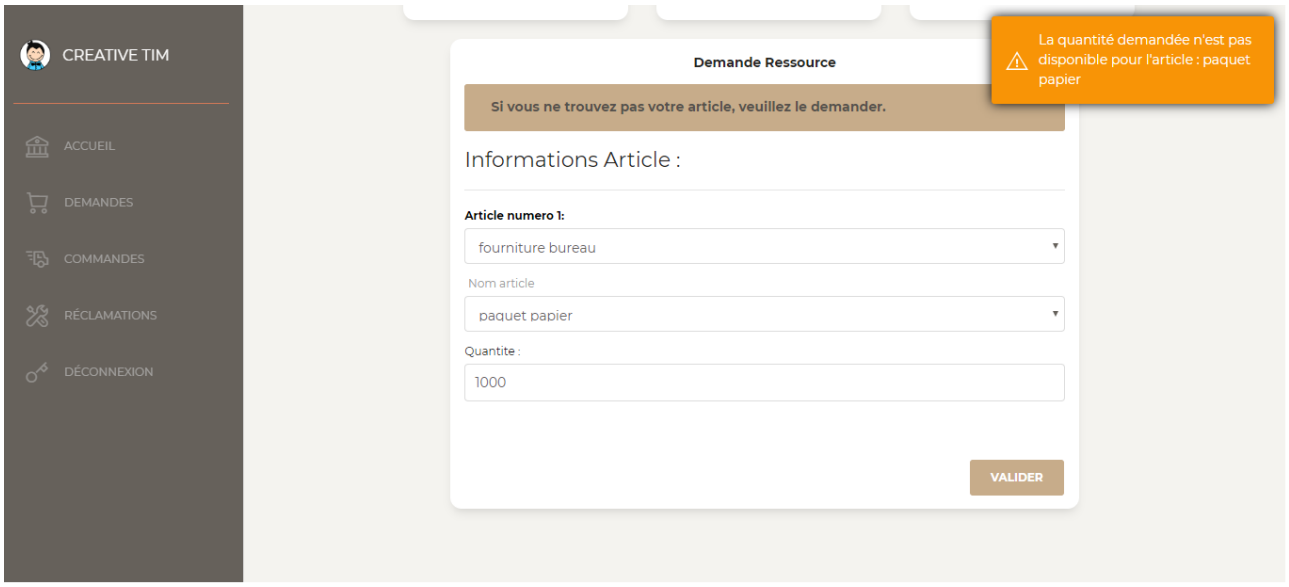

Figure 34 : Message d'erreur – Quantité indisponible -

- Listes des demandes

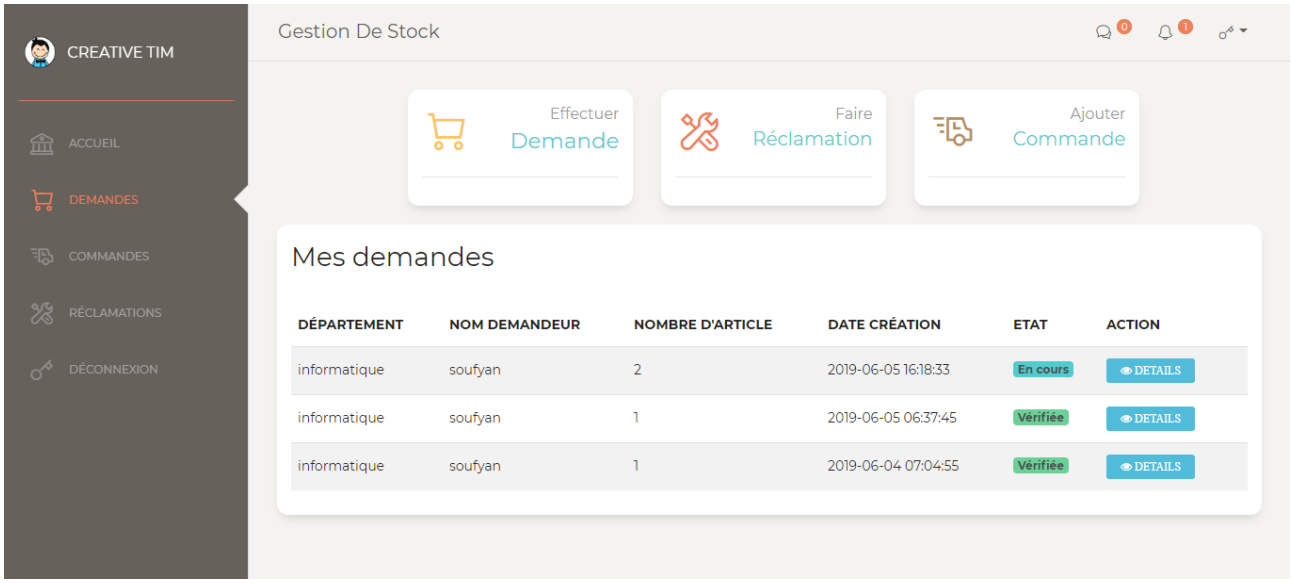

Figure 35 : Liste des demandes

- Notification quand l'article commandée est disponible au stock

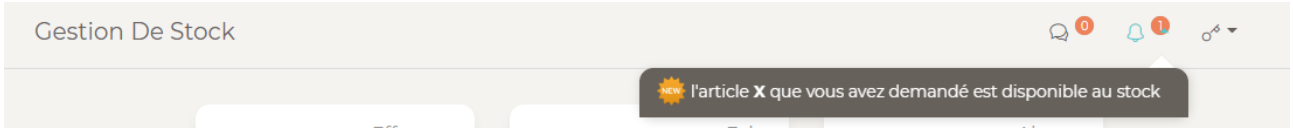

Figure 36 : Notification - Article commandé est disponible -

2.3. Interfaces du chef de département

- Accueil

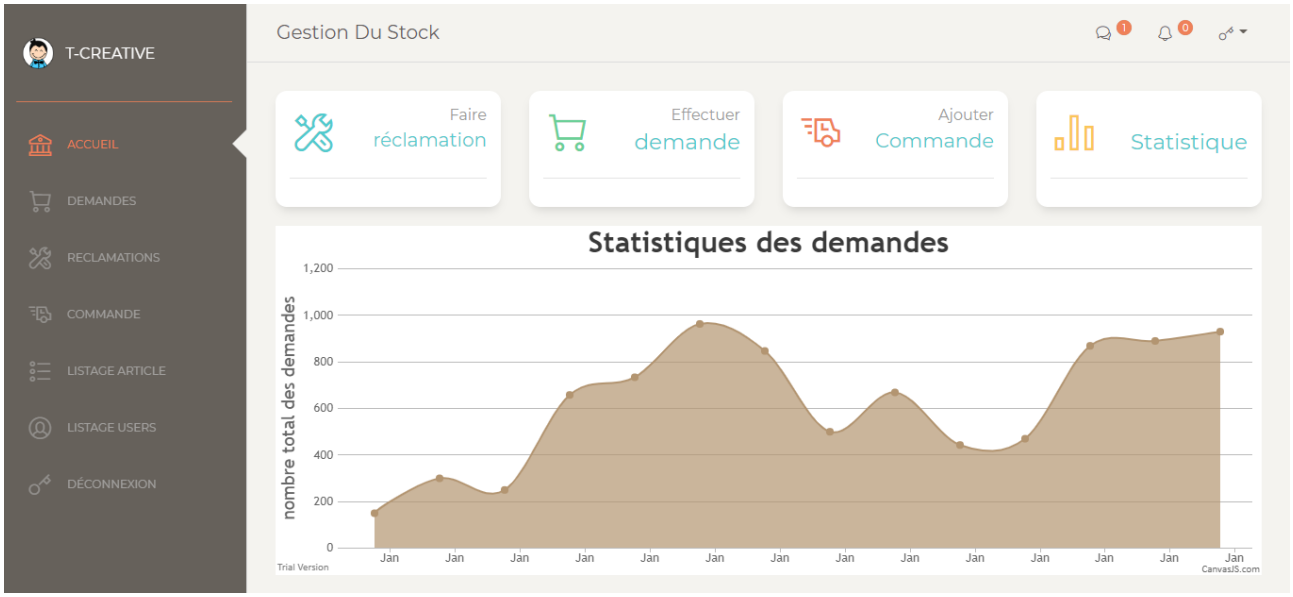

Figure 37 : Fenêtre d'accueil du chef de département

- Choisir une option

Une fenêtre d'option apparait au chef du département à chaque fois il veut accéder aux demandes ou aux réclamations.

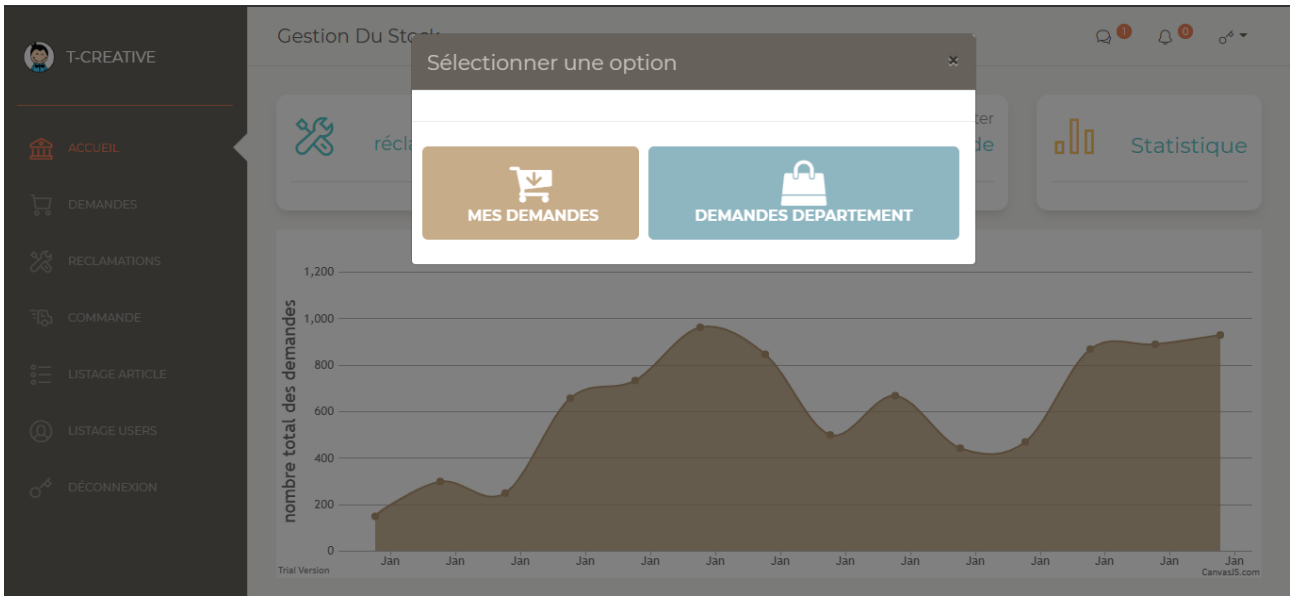

Figure 38 : Fenêtre - Sélectionner option -

- Liste des demandes du département

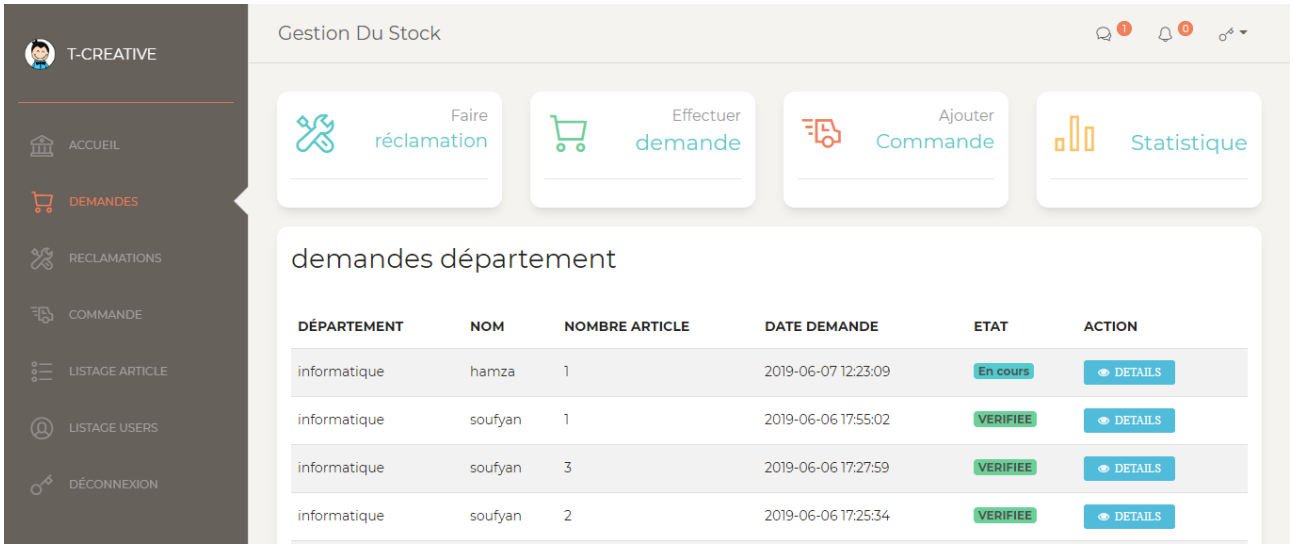

Figure 39 : Liste des demandes du département

- Liste des réclamations du département

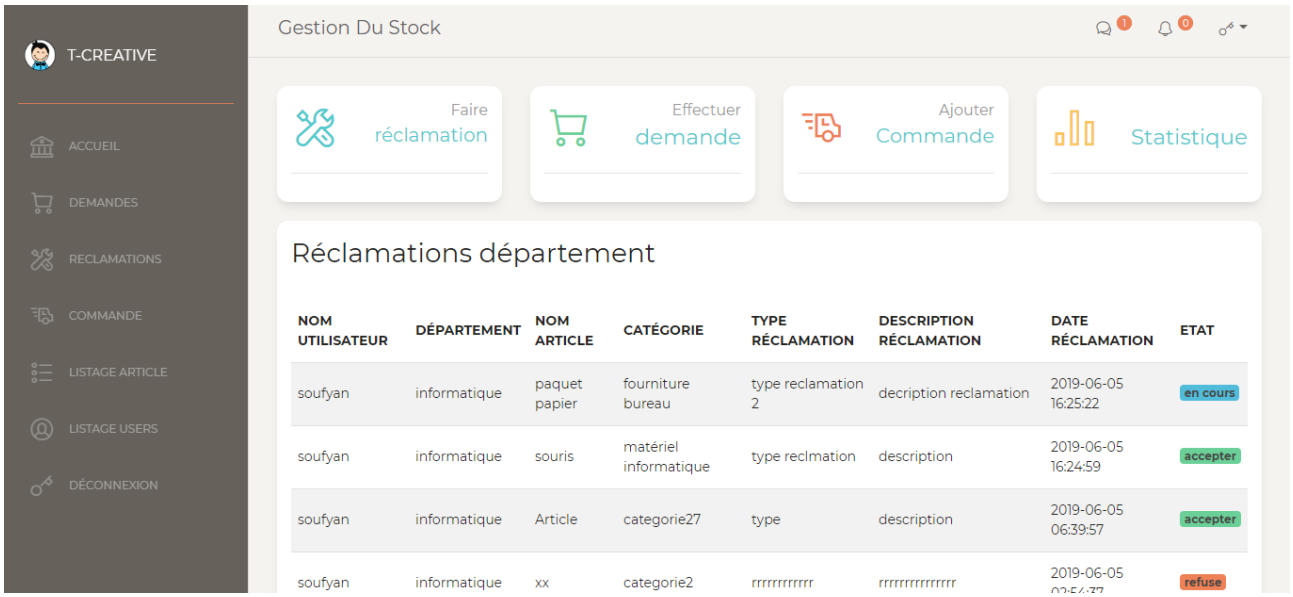

Figure 40 : Liste des réclamations du département

Conclusion et perspectives

Le présent projet réalisé au cours de notre stage de fin de formation a été très bénéfique tant au niveau personnel qu'au niveau académique. Ainsi, il nous a permis de consolider nos connaissances en termes d'outils de programmation et de conception, ainsi que la gestion de projets. Il a été une opportunité pour développer de nouvelles compétences dans la programmation avec PHP et maitriser le Framework Laravel qui est un Framework très complet et très puissant.

Ce stage nous a aussi offert la chance de découvrir et travailler dans un contexte réel et affronter de vraies problématiques. Ces dernières nous ont aidés à acquérir une expérience très riche et diversifiée. Ainsi, on a travaillé avec une base de données réelle et volumineuse en termes de tables et relations avec tout ce que cela implique en termes de complexité du projet.

La réalisation de ce projet avait comme objectif l'automatisation de la procédure de demande des ressources matérielles dans les établissements universitaires, ainsi que la simplification et l'amélioration du travail administratif du responsable du stock en réalisant une application qui donne la possibilité de gérer les demandes, les réclamations reçues, ainsi que pour contrôler et traiter les mouvements du stock.

Cette application reste toujours extensible par d'autres développeurs pour d'autres améliorations et ajout de nouvelles fonctionnalités ou aussi pour l'entretenir et l'adapter aux besoins émergents.

Webographie

• Documentation Laravel :

https://laravel.com/docs/5.8

• Documentation Bootstrap :

http://www.w3schools.com

https://mdbootstrap.com/docs/

• Forum & Astuces :

http://stackoverflow.com

https://laravel.fr/

• Cours en ligne :

https://openclassrooms.com/courses/dynamisez-vos-sites-web-avec-javascript

https://openclassrooms.com/courses/simplifiez-vos-developpements-javascript-avec-jquery

https://openclassrooms.com/courses/decouvrez-le-framework-php-laravel-1

• Tutoriels :

https://www.youtube.com/playlist?list=PLNAMnK22kwLajj5F8T_00t6IRDUpkp0y_

https://www.youtube.com/playlist?list=PLe30vg_FG4OTO7KbQ6TByyY99AiSw1MDS